S586s

Silva, Renan Fischer e
Sistemas de gerenciamento de chaves públicas baseado em virtualização para redes AD HOC móveis [manuscrito] / Renan Fischer e Silva. – Curitiba, 2010.
103 f. . : il. [algumas color.] ; 30 cm.

Impresso.
Dissertação (mestrado) - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-graduação em Informática , 2010.

Orientador: Luis Carlos Pessoa Albini

1.Redes de computação. 2. Sistema de gerenciamento (VKM-Virtual Key Management). I. Universidade Federal do Paraná. II. Albini, Luis Carlos Pessoa. III. Título.

CDD: 004.6

CRB/9 - 1585

RENAN FISCHER E SILVA

SISTEMA DE GERENCIAMENTO DE CHAVES PÚBLICAS BASEADO EM VIRTUALIZAÇÃO PARA REDES AD HOC MÓVEIS

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.
Orientador: Prof. Dr. Luiz Carlos Pessoa Albini

CURITIBA

2010

RENAN FISCHER E SILVA

SISTEMA DE GERENCIAMENTO DE CHAVES PÚBLICAS BASEADO EM VIRTUALIZAÇÃO PARA REDES AD HOC MÓVEIS

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.
Orientador: Prof. Dr. Luiz Carlos Pessoa Albini

CURITIBA

2010

RENAN FISCHER E SILVA

SISTEMA DE GERENCIAMENTO DE CHAVES PÚBLICAS BASEADO EM VIRTUALIZAÇÃO PARA REDES AD HOC MÓVEIS

Dissertação aprovada como requisito parcial à obtenção do grau de Mestre no Programa de Pós-Graduação em Informática da Universidade Federal do Paraná, pela Comissão formada pelos professores:

Orientador: Prof. Dr. Luiz Carlos Pessoa Albini
Departamento de Informática, UFPR

Prof. Dr. Augusto Foronda
Departamento de Eletrônica e Telecomunicações, UTFPR

Profa. Dra. Michele Nogueira
Departamento de Informática, UFPR

Curitiba, 27 de agosto de 2010

Para meus amados pais

AGRADECIMENTOS

Primeiramente agradeço a Deus por ter chego ao fim deste trabalho. Sem Ele, nada disso teria sido possível. Obrigado pela saúde encontrada durante o tempo de elaboração do mesmo e principalmente pela capacidade concedida para finalizá-lo.

Aos meus pais, por terem me proporcionado sempre o melhor que podiam oferecer em termos de educação, conforto, amor e carinho. Por muitas vezes terem se sacrificado para me dar condições de sonhar mais do que eles sonharam quando tinham a minha idade. Pelo exemplo, por estarem sempre presente e acima de tudo, por nunca terem deixados de serem espelhos durante a construção da pessoa que me tornei.

Agradeço muito ao meu orientador, Prof. Albini. Obrigado por todo o aprendizado adquirido, não somente o ensinado nas salas de aulas, mas também aquele que usarei durante a vida. Por ter paciência nos momentos necessários e mais ainda, por cobrar nos momentos cruciais. Principalmente, por ter continuado a me orientar durante a etapa final da elaboração deste trabalho e pelas noites que abriu mão de ficar com sua família para discutir detalhes da dissertação com seu orientado que se encontrara a quilômetros de distância.

Um agradecimento especial vai para uma pessoa muito importante para mim. Meu irmão gêmeo Alan. Obrigado por todo o apoio e todo o suporte ao fazer dessa nova fase da minha vida um caminho mais tranqüilo a ser trilhado. Com certeza sem a sua ajuda as coisas teriam sido muito mais difíceis.

Finalmente, não posso deixar de agradecer ao grupo NR2, seus pesquisadores e meus colegas. Obrigado pela infra-estrutura concedida para simulações. Em especial ao colega Eduardo da Silva, que participou dos resultados obtidos nesse trabalho compartilhando os resultados obtidos durante o seu mestrado e trocando idéias para implementações e soluções para diversos problemas encontrados durante esse percurso.

Pouco conhecimento faz com que as pessoas se sintam orgulhosas. Muito conhecimento, que se sintam humildes.

Leonardo di ser Piero **da Vinci**

RESUMO

MANETs (*Mobile Ad Hoc Networks*) são redes sem fio e sem infra-estrutura estabelecidas dinamicamente, sem a necessidade de uma administração centralizada. Devido ao roteamento distribuído nessas redes e ao meio de comunicação sem fio redes Ad Hoc podem apresentar todos os problemas de segurança existentes em redes convencionais e ainda novos desafios. O uso de criptografia é a principal técnica para garantir a transferência de dados em uma rede de forma segura. Nos sistemas criptográficos assimétricos, os nós utilizam uma chave para cifrar uma mensagem e outra chave para decifrar a mesma. A tarefa de administrar essas chaves é realizada por um Sistema de Gerenciamento de Chaves, que define a emissão, o armazenamento, a distribuição, a proteção e a revogação das mesmas. Esse trabalho apresenta um novo Sistema de Gerenciamento de chaves baseado em Virtualização. Nesse sistema, chamado de *Virtual Key Management* (VKM), é utilizado uma estrutura virtual, sem qualquer relação com as coordenadas físicas dos nós da rede, para estabelecer a confiança entre os mesmos. Dessa forma, os nós seguem as regras estabelecidas por essa estrutura para realizar a emissão, o armazenamento, a distribuição, a proteção e a revogação de chaves públicas e de chaves privadas na rede. O VKM é 100% resistente a ataques de Criação de Identidades Falsas. Quando comparado com o Sistema de Gerenciamento de Chaves Públicas Auto-organizado (PGP-Like), o VKM mostra maior resistência contra ataques de Personificação e a mesma resistência contra ataques de Falta de Cooperação. Quando comparado com o *Group-based Key Management* (GKM), o VKM mostra maior resistência contra ataques de Criação de Identidades Falsas por ser 100% resistente ao mesmo. O *Virtual Routing Protocol* (VRP) e o *Virtual Distance Vector* (VDV) são dois protocolos de roteamento híbridos que utilizam uma estrutura virtual para definir a parte pró-ativa do protocolo. Esse trabalho também mostra que o impacto no roteamento causado pela incorporação do VKM nesses protocolos de roteamento causa queda na taxa de entrega de dados, aumento do atraso no envio de mensagens e aumento da sobrecarga gerada na rede.

ABSTRACT

MANETs (Mobile Ad Hoc Networks) are wireless networks without infrastructure that are dynamically established, without the need of a centralized administration. Due to distributed routing in these kind of networks and also due to wireless media, Ad Hoc networks may present all security problems existing in conventional networks and also new and challenging security problems. The usage of cryptography is the main technique to assure the secure data transferring in a network. In asymmetric cryptographic systems, nodes use a key to code a message and another key to decode a same. The task of managing these keys is performed by a Key Management System, that defines the issue, the storage, the distribution, the protection and the revocation of them. This work presents a new Key Management System based on Virtualization. In this system, called Virtual Key Management System (VKM), is used a virtual structure, without any relation to the physical coordinates of the nodes, to establish the confidence between themselves. Therefore, nodes follow the rules established by this virtual structure to perform the issue, the storage, the distribution, the protection and the revocation of the public and the private keys in the network. The VKM is 100% resilient against Creation of Fake Identity attacks. When comparing to the Self-Organized Public Key Management (PGP-Like), the VKM shows better resiliency against attacks of Personification and the same resiliency against attacks of Lack of Cooperation. When comparing to the Group-based Key Management (GKM), the VKM shows higher resiliency against Creation of Fake Identity attacks, once it is 100% resilient to this kind of attack. The Virtual Routing Protocol (VRP) and the Virtual Distance Vector (VDV) are two hybrid routing protocols that makes usage of a virtual structure to define the proactive part of the routing. This work also shows that the impact created with the implementation of VKM in these routing protocols are the decrease of the data delivery ratio, the increase of the delay and the increase of the network overhead.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AODV Ad Hoc On-Demand Distance Vector

ARAN Authenticated Routing for Ad Hoc Networks

CBR Constant Bit Rate

CCC Ciclos de Cubos Conectados

CE Certificate Exchange Convergence

CSN Chains with Sybil Nodes

DBF Distributed Bellman-Ford

DSDV Destination-Sequenced Distance-Vector

DSR Dynamic Source Routing

DV Distance Vector

IEEE Institute of Electrical and Electronics Engineers

IETF Internet Engineering Task Force

MANET Mobile Ad-hoc Networks

PGP Pretty Good Privacy

PGP-Like Self-Organized Public Key Management

RERR Route Error

RFC Request for Comments

RREP Route Reply

RREQ Route Request

RTRANS Route Translation

SAODV Securing Ad Hoc Routing Protocols

SEAD Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks

SECRYPT International Conference on Security and Cryptography

SLSP Secure Link State Routing for Mobile Ad Hoc Networks

SRAN Secure Routing Protocol for Wireless Ad Hoc Networks

SYN Synchronize

TCP Transmission Control Protocol

TPDR Tabela de Pacotes de Dados Recentes

UR User Reacheability

VDV Virtual Distance Vector

VKM Virtual Key Management

VKM-PA VKM with Proactive Authentication

VKM-RA VKM with Reative Authentication

VRP Virtual Routing Protocol

WTF Workshop on Test and Fault Tolerance

ZRP Zone Routing Protocol

NOTAÇÃO

u	identidade do dispositivo (nó) u
K_u	chave pública do nó u
prK_u	chave privada do nó u
$(v, Kv)prKu$	certificado emitido por u que associa K_v a v e assina com prK_u
$K_u \rightarrow K_w$	ato de emitir um certificado assinado por u que amarra K_w a w
$K_u \rightsquigarrow K_w$	caminho de certificados(cadeia de certificados) que conecta K_u a K_w
G	grafo que representa a estrutura virtual nos protocolos VRP e VDV
V	conjunto de unidades da rede nos protocolos VRP e VDV
A	conjunto de enlaces virtuais da rede nos protocolos VRP e VDV
P_u	conjunto formado por todas as unidades que são espiãs de u
S_u	conjunto formado por todas as unidades que são espionadas por u
N_j	conjunto de vizinhos de d
L	grafo usado para representar a estrutura virtual no VKM
D	conjunto de unidades da rede no VKM
E	conjunto de arestas que corresponde aos enlaces virtuais da rede no VKM
L_u	repositório local de certificados não atualizados de um nó u
L_u^N	repositório local de certificados atualizados de um nó u
r_n	n-ésimo elemento de uma rota formada por n nós
α	unidade de *broadcast* na rede
δ	deslocamento de tempo para próxima descoberta de rotas na rede
T_v	tempo de vida de um certificado na rede
S	grau dos vértices da estrutura virtual usada no VKM, no VRP e no VDV

CAPÍTULO 1

INTRODUÇÃO

Mobile Ad Hoc Networks (MANETs) são redes sem fio e sem infra-estrutura. Essas redes são formadas por dispositivos (nós) que se comunicam entre si usando um meio de comunicação sem fio. As redes Ad Hoc são estabelecidas dinamicamente, sem a necessidade de uma administração centralizada. Além de não possuírem uma infra-estrutura fixa, elas devem permitir a entrada e a saída de nós da rede, pois devido a mobilidade, os mesmos podem entrar e sair da rede dinamicamente. Como exemplo de aplicações para MANETs, destacam-se resgates de emergência, sensores digitais, comunicação em campos de batalha e compartilhamento de dados durante uma conferência [12].

Nas redes Ad Hoc, os nós se comunicam através de ondas de rádio. Dessa forma, possuem alcance limitado e, quando não conseguem alcançar um determinado destino, devem encaminhar o pacote para um nó vizinho, de modo a ser transmitido sucessivamente até o destinatário final da mensagem. Para o funcionamento dessas redes, a tarefa de roteamento de mensagens deve ser delegada aos próprios nós da rede de forma cooperativa, distribuída e auto-organizável [27, 24, 47, 37]. Para a tarefa de rotear os pacotes de dados em redes Ad Hoc, vários protocolos já foram desenvolvidos, como: *Ad Hoc On-Demand Distance Vector Routing* (AODV) [37], *Dynamic Source Routing* (DSR) [27], *Destination-Sequenced Distance Vector Routing* (DSDV) [36], *Virtual Routing Protocol* (VRP) [2] e *Zone Routing Protocol* (ZRP) [21]. Esses protocolos em geral podem ser classificados como reativos, pró-ativos, híbridos, geográficos, virtuais, entre outros.

Algumas aplicações para MANETs podem conter uso de transações comerciais, informações industriais e trocas de dados corporativos de uma empresa. Essas aplicações podem ser usadas em campos militares ou mesmo em ambientes onde é necessário uma rede auto-configurável [20]. Nesse tipo de ambiente, essas redes podem sofrer diversos tipos de ataques [46]. Devido ao roteamento distribuído nessas redes e ao meio de

comunicação sem fio, nós mal-intencionados podem facilmente interceptar informações (sigilosas ou não). Tal ato pode ser caracterizado como um ataque passivo. Além de apenas interceptar informações, nós maliciosos podem alterar o funcionamento da rede, nesse caso caracterizando-se como um ataque ativo [31, 46]. Redes móveis são, portanto, mais suscetíveis a ameaças do que redes cabeadas [3].

Considerando o uso do meio de comunicação sem fio e a topologia dinâmica das MANETs, a segurança é um dos maiores desafios nesse tipo de rede. Devido a tais características, as redes Ad Hoc podem apresentar todos os problemas de segurança existentes em redes convencionais e ainda novos desafios [3]. A topologia dinâmica das MANETs facilita a ação de nós mal-intencionados, tornando-as susceptíveis a diferentes tipos de ataques como *blackhole*, *wormhole*, ataques de falta de cooperação e ataques *Sybil*)[12]. Neste trabalho serão abordados os ataques de falta de cooperação e ataques do tipo *Sybil*. Os ataques do tipo *Sybil* serão divididos em dois tipos diferentes: criação de identidades falsas ou personificação de uma identidade existente na rede.

A criptografia é considerada a principal técnica para garantir a transferência de dados de forma segura na rede [1]. Para o uso de criptografia é necessária a utilização de chaves relacionadas aos nós da rede. Os sistemas de criptografia podem ser classificados em simétricos e assimétricos. Nos sistemas simétricos os nós utilizam a mesma chave para cifrar e decifrar mensagens. Nos sistemas assimétricos, os nós utilizam uma chave para cifrar uma mensagem e outra chave decifrar a mesma [14]. A tarefa de administrar essas chaves é realizada por um esquema de gerenciamento de chaves, que define a emissão, o armazenamento, a distribuição, a proteção e a revogação das mesmas [12].

Os esquemas de gerenciamento de chaves para redes *Ad Hoc* móveis precisam funcionar em ambientes com topologia dinâmica e ser auto-organizáveis e descentralizados [9, 22, 41, 43]. Além disso, devem satisfazer alguns requisitos como: (*i*) não pode haver um ponto de único falha que comprometa a integridade da rede; (*ii*) devem ser tolerantes a falhas, pois o comprometimento de um certo número de nós não deve afetar a segurança entre os nós não comprometidos; (*iii*) devem ser capazes de revogar as chaves dos nós comprometidos de maneira eficiente e segura e também de atualizar as chaves dos nós

não-comprometidos; (*iv*) devem ser eficientes em termos de armazenamento, computação e comunicação dos dados.

É possível classificar os esquemas de gerenciamento de chaves para MANETs em [15]: baseado em identidade [29], baseado em cadeias de certificados [7, 9, 25], baseado em clusters [32, 33], baseado em pré-distribuição [19] e baseado em mobilidade [8]. Entre estes, os esquemas baseados em certificados são os que melhor se adaptam em ambientes MANETs [12], por serem totalmente distribuídos e auto-organizados e não necessitarem de qualquer tipo de entidade central nem mesmo na fase de formação da rede [12]. O principal esquema de gerenciamento de chaves totalmente distribuído e auto-organizável é o *Self-Organized Public Key Management System* [7, 25]. Esse sistema de gerenciamento de chaves será referenciado nesse trabalho por *PGP-Like* seguindo referência adotada em [41]. O esquema de gerenciamento de chaves baseado em grupos - *Group-based Key Management* (GKM) também será abordado neste trabalho.

O *PGP-Like* é um esquema de gerenciamento de chaves auto-organizável baseado nos conceitos do PGP [48], no qual todo par de chaves pública e privada é criado pelos próprios nós da rede. Os nós também emitem certificados para outros nós em que confiam. Cada nó tem um repositório local de certificados que é periodicamente trocado com seus vizinhos. Chaves são autenticadas através de cadeias de certificados, construídas usando o repositório local dos nós.

No GKM [34], usuários formam pequenos grupos, nos quais todos os nós tem o mesmo papel sem a necessidade de haver um líder. Estes grupos são formados com base no relacionamento dos usuários que formam uma rede de grupos. Essa rede é usada para realizar todas as operações de gerenciamento de chaves e os nós precisam ser um membro de um grupo para fazer parte do sistema.

1.1 Ataques em MANETs

Nas MANETs, os nós maliciosos podem tanto tirar proveito do meio de comunicação sem fio escutando uma mensagem em modo promíscuo (ataque passivo), como alterar o roteamento quando o mesmo estiver roteando pacotes (ataque ativo). Dessa forma, os

mesmos podem roubar informação sigilosa da rede e também alterar o roteamento para prejudicar o funcionamento da mesma [31, 46].

Esses tipos de ataques podem ser definidos como ataques passivos e ativos [31]. Ataques passivos são aqueles onde o atacante não ocasiona danos nem alterações ao funcionamento da rede. Os atacantes estão apenas interessados em obter informações valiosas escutando o tráfego da rede. Neste caso, devido às características particulares de um ambiente de rede sem fio, a tarefa de detectar este tipo de invasor torna-se praticamente impossível [46].

Os ataques ativos podem ser classificados em: ataques que afetam um serviço da rede (protocolos de roteamento ou sistemas de gerenciamento de chaves por exemplo) e ataques que afetam os recursos da rede [24]. Os ataques que afetam serviços da rede por exemplo fazem com que pacotes legítimos sejam roteados de maneira incorreta ou nós sejam autenticados de erroneamente. Já os ataques que afetam os recursos da rede são realizados introduzindo novos pacotes, de forma a consumir recursos da rede como banda, ou mesmo recursos dos nós como memória ou bateria [3]. Para realizar um ataque ativo, o atacante deve modificar o conteúdo das mensagens por ele roteadas ou injetar propositadamente pacotes na rede. A Tabela 1.1 foi baseada naquela apresentada por Bannack *et.al* [5] e classifica os tipos de ataques quanto à camada onde ocorrem. Alguns tipos de ataques como ataques à camada de rede, interferem diretamente na performance do roteamento da rede [46]. Alguns desses ataques podem ser evitados protegendo a informação que se está roteando com o uso de criptografia [20].

1.1.1 Ataques ao Sistema de Gerenciamento de Chaves Públicas Auto-organizado

Os ataques aos sistemas de gerenciamento de chaves em MANETs visam comprometer a disponibilidade, confidencialidade, integridade, autenticidade e irretratabilidade do mesmo[12]. Dentre todos os diferentes tipos de ataques que podem ser feitos nas MANETs, dois tipos de ataques podem ter um grande efeito se aplicados em um sistema de gerenciamento de chaves [3]: ataques de falta de cooperação e ataques *Sybil*.

Tabela 1.1: Tipos de ataques em MANETs

Camada	Ataque	Descrição
Física	Ruído	interferência no sinal transmitido, negando o serviço no canal de comunicação.
Enlace	Colisão	colisões propositais causadas por um atacante, com o objetivo de negar o serviço no canal de comunicação.
Rede	*Wormhole*	os nós maliciosos cooperam entre si para fornecer um canal paralelo de baixa latência para a comunicação.
	Blackhole	o nó malicioso exclui todos os pacotes que chegam a ele para ser roteados.
	Grayhole	uma variação do blackhole, na qual o atacante descarta alguns pacotes seletivamente, dificultando a sua detecção.
	Sinkhole	o atacante força os pacotes a passar por um determinado nó, facilitando a ação de outros ataques.
	Aceleração	nós maliciosos encaminham rapidamente as mensagens de pedido de rota, com o objetivo de participar dessa rota.
	Envenenamento da tabela de roteamento	nós maliciosos fabricam mensagens com rotas falsas de forma que todas as unidades da rede que estejam em modo de escuta promíscua tenham sua tabela de rotas envenenada.
	Divulgação da localização	atacantes revelam informações relacionadas à localização dos nós da rede ou relacionadas à estrutura da rede.
	Blackmail	relacionado à protocolos que utilizem uma *black list*, atacantes podem fabricar falsos relatórios de forma a desconectar nós íntegros do restante da rede.
Transporte	Inundação de SYN	inundação clássica de pacotes TCP SYN, na qual um atacante envia muitos pedidos de estabelecimento de conexão a um outro nó, sobrecarregando os recursos desse nó.
Multi-camadas	Exaustão ou inundação	tentativas de retransmissões sucessivas, com o objetivo de derrubar os recursos limitados da vítima, como processador, memória, bateria ou largura de banda.
	Sybil	os nós maliciosos criam várias identidades falsas.
	Falta de cooperação	os nós egoístas comprometem o funcionamento da rede não cooperam com as suas atividades.
	Replay	nós maliciosos injetam na rede tráfego capturado por eles previamente, podendo prejudicar protocolos mal projetados.

Durante um ataque de **falta de cooperação**, nós optam por não cooperar com o funcionamento da rede. Em redes Ad Hoc, a cooperação mútua entre os nós da rede é primordial para o funcionamento com topologia dinâmica e para os nós da rede conseguirem realizar operações na mesma. O mal funcionamento de alguns nós que optem por não cooperar com o funcionamento da rede compromete o desempenho e a eficácia desse sistema.

Durante um ataque **Sybil**, nós mal intencionados criam múltiplas identidades apesar de usarem um único dispositivo físico [16]. As identidades adicionais na rede podem ser obtidas quando o nó malicioso **cria uma nova identidade** ou **rouba a identidade de outro nó** pertencente a rede, **personificando-o** [16]. Em sistemas que confiam na redundância de informações, deve-se garantir a unicidade das identidades dos nós da rede [25]. Como esse ataque ocorre em diversas camadas de rede [46], é possível reduzir a eficácia de:

1. esquemas com armazenamento distribuído [13].

2. roteamento multicaminhos [30].

3. mecanismos baseados em confiança ou em eleição [42].

4. esquemas de reputação [10].

O ataque *Sybil* também pode ser usado para comprometer um sistema de gerenciamento de chaves em MANETs [45]. Diversos sistemas de gerenciamento de chaves para redes Ad Hoc utilizam certificação em grupo ou cadeias de certificados para amenizar os requisitos de uma unidade certificadora centralizada de chaves públicas [12]. Um atacante que possua grande quantidade de identidades distintas na rede pode controlar efetivamente um esquema de gerenciamento de chaves, e dessa forma permitir a entrada de outros nós no sistema [39].

1.2 Motivação

Como mostrado em [41], o *PGP-Like* é altamente vulnerável a ataques *Sybil* [16]. Ataques *Sybil* consistem em um atacante usar identidades falsas para enganar, alterar e/ou prejudicar o funcionamento dos protocolos da rede [16]. A funcionalidade do *PGP-Like* é comprometida mesmo quando apenas 5% de nós falsos estão presentes na rede.

Já o GKM é resistente a ataques *Sybil* em 90% das ocorrências [16]. Entretanto o GKM ainda pode ser comprometido por ataques de criação de falsa identidade. Neste tipo de ataque o GKM deixa de ser resistente a partir de 20% de atacantes [12].

O objetivo deste trabalho é introduzir um novo esquema de gerenciamento de chaves públicas para MANETs, o *Virtual Key Management System* (*VKM*), mais resistente a um tipo específico de ataque *Sybil*: o ataque de **criação de identidades falsas**. O VKM usa uma estrutura virtual para indicar a confiança entre os nós e a formação das cadeias de certificados. Estruturas virtuais já foram usadas em Redes Ad Hoc para aumentar a taxa de entrega [2] e também para diminuir o atraso no envio de dados [40] ao serem combinadas com protocolos de roteamento. É possível obter virtualmente 100% de segurança contra os ataques de criação de falsa identidade ao se utilizar uma estrutura virtual como uma rede *overlay* sobre a topologia física dos nós. Apenas os nós que realmente fizerem parte da rede conhecem as regras estabelecidas logicamente por essa estrutura. Dessa forma,

um nó malicioso que deseje personificar diversas identidades não faz parte da rede *overlay* definida pela estrutura virtual, e como essa estrutura virtual é fixa, a mesma não permite a entrada de nós na rede. Apenas no caso onde todos os nós da rede aceitem remodelar a esturtura virtual seria possível permitir a entrada de novos nós na rede. Dessa forma, um atacante ficaria impossibilitado de criar novas identidades na rede a não ser que conseguisse convencer todos os nós que fazem parte dessa rede *overlay* a remodelar a estrutura virtual.

No VKM, a estrutura virtual é estática e conhecida por todos os nós. Portanto, este esquema não permite a entrada de novos nós na rede facilmente. Esta característica pode não ser desejável em ambientes MANETs, onde a entrada de novos nós na rede é muito comum. Entretanto, há situações onde é mais fácil prever os nós que farão parte da rede como comunicações privadas, onde os participantes são conhecidos desde o começo da comunicação. Adicionalmente, o VKM permite que os nós saiam do sistema a qualquer momento, afetando apenas a quantidade de cadeias de certificados válidas disponível na estrutura virtual.

A estrutura virtual é altamente redundante e não possui relação com a localização física das unidades da rede. Estruturas virtuais já foram usadas em protocolos de roteamento para redes Ad Hoc como no *Virtual Routing Protocol* (VRP) [2] e no *Virtual Distance Vector* (VDV) [40]. Além de poder ser implementado em qualquer tipo de aplicação para prover segurança, o VKM é facilmente implementado em qualquer protocolo de roteamento. Como o VRP e o VDV já fazem uso de uma estrutura virtual, a implementação do VKM torna-se uma tarefa fácil. Porém, para se fazer possível utilizar o VKM incorporado diretamente nesses protocolos de roteamento seria preciso antes realizar um estudo sobre o impacto do VKM quando implementado diretamente nos respectivos protocolos. Esse estudo também foi uma motivação para esse trabalho, e será apresentado na presente dissertação.

1.3 Objetivos

Nesse trabalho é apresentado um novo esquema de gerenciamento de chaves para MANETs, que quando comparado com o *PGP-Like*, apresenta-se mais robusto contra ataques do tipo *Sybil*). Adicionalmente, quando comparado com o GKM, o VKM apresenta total segurança contra o ataque de criação de identidades falsas. O esquema de gerenciamento de chaves *Virtual Key Management* (VKM)[18, 17] é baseado em certificados [7, 25], é distribuído e é auto-organizável.

O VKM é o primeiro esquema de gerenciamento de chaves baseado completamente em virtualização. Para garantir a segurança contra ataques do tipo *Sybil*, principalmente ataques de criação de falsa identidade, o VKM faz o uso de uma estrutura virtual que estabelece a confiança entre os nós da rede, definindo a emissão de certificados pelos mesmos.

O VKM encaixa-se perfeitamente no funcionamento de protocolos de roteamento que façam uso de uma estrutura virtual, relacionada ou não com a topologia física da rede. Por outro lado, devido as características particulares de cada protocolo de roteamento, é desejado conhecer o impacto no desempenho (taxa de entrega, atraso no envio de dados e sobrecarga) causado pelo VKM ao ser incorporado por diferentes protocolos de roteamento virtuais.

Esse trabalho apresentará as seguintes contribuições:

1. um novo sistema de gerenciamento de chaves VKM, que ao fazer uso de estruturas virtuais para o gerenciamento das chaves públicas e privadas da rede, seja resistente a ataques do tipo *Sybil*.

2. análise da comparação entre o VKM e o *PGP-Like* em cenários de falta de cooperação e com ataques de personificação.

3. análise da comparação entre o VKM e o *GKM* em cenários com ataques de personificação e com ataques de criação de identidades falsas.

4. implementação do VKM nos protocolos de roteamento que fazem uso de uma estru-

tura virtual como VRP [2] e o VDV [40].

5. estudo do impacto no desempenho dos protocolos de roteamento VRP [2] e o VDV [40] ao implementar o VKM nesses protocolos. Esse estudo consiste das análises da taxa de entrega de pacotes de dados, do tempo de envio de dados e da sobrecarga do roteamento causada na rede.

1.4 Estrutura e organização

Este trabalho está organizado em 7 capítulos. O Capítulo 2 resume os esquemas de gerenciamento PGP-Like e GKM e apresenta os resultados desses sistemas de gerenciamento de chaves quando submetidos a ataques como os ataques de falta de cooperação e os ataques do tipo *Sybil*.

O Capítulo 3 apresenta o novo sistema de gerenciamento de chaves. É apresentada a visão geral do funcionamento, assim como o funcionamento detalhado do VKM operando em modo reativo e do VKM operando em modo pró-ativo.

O Capítulo 4 apresenta a análise e comparação dos resultados entre o VKM e o *PGP-Like* quando submetidos a cenários de falta de cooperação e a ataques de personificação, os resultados entre o VKM e o GKM quando submetidos a ataques de personificação e os resultados do VKM e do GKM quando submetidos a ataques de criação de falsa identidade.

O Capítulo 5 oferece um resumo dos protocolos de roteamento existentes. Os protocolos de roteamento existentes são classificados em diferentes tipos como reativos, pró-ativos e híbridos. Por fim, são apresentados os protocolos de roteamento virtuais: VRP e VDV.

O Capítulo 6 apresentada a análise do impacto do desempenho do VKM quando incorporado aos protocolos de roteamento virtuais VRP e VDV.

Finalmente, o Capítulo 7 apresenta as conclusões obtidas com esse trabalho, além de apresentar uma análise dos resultados obtidos, e uma relação dos trabalhos futuros que darão sequência ao trabalho aqui apresentado.

CAPÍTULO 2

SISTEMAS DE GERENCIAMENTO DE CHAVES

Os sistemas de gerenciamento de chaves públicas são responsáveis por administrar as chaves públicas e privadas de uma rede quando se utiliza criptografia assimétrica. A administração consiste em emitir, em armazenar, em distribuir, em proteger e em revogar as chaves públicas e privadas da rede [12].

Esquemas de gerenciamento de chaves para MANETs precisam funcionar em ambientes com topologia dinâmica, ser auto-organizáveis e descentralizados [9, 22, 41, 43]. É possível classificar os esquemas de gerenciamento de chaves para MANETs em [15]: baseado em identidade [29], baseado em cadeias de certificados [7, 9, 25], baseado em clusters [32, 33], baseado em pré-distribuição [19] e baseado em mobilidade [8]. Entre estes, os esquemas baseados em certificados são os que melhor se adaptam em ambientes MANETs, por serem totalmente distribuídos e auto-organizados e não necessitarem de qualquer tipo de entidade central nem mesmo na fase de formação da rede [12].

2.1 PGP-Like

O *PGP-Like* é o principal [7, 25] esquema de gerenciamento de chaves públicas para MANETs totalmente distribuído e auto-organizável [7, 25]. As chaves públicas e privadas dos nós são criadas pelos próprios nós seguindo os conceitos do PGP [48]. Além disso, cada nó emite certificados de chave pública para outros nós nos quais confia.

No *PGP-Like*, se um nó u acredita que uma dada chave pública K_v pertence a um dado nó v, ele pode emitir um certificado amarrando K_v ao nó v, $(v, K_v)_{prK_u}$, onde prK_u é a chave privada do nó u. Este certificado é armazenado no repositório local de certificados de u e de v. Adicionalmente, cada nó periodicamente realiza trocas de seu repositório com seus vizinhos físicos.

As chaves públicas e os certificados são representados por um grafo dirigido $L =$

(D, E), no qual D representa as chaves públicas e E representa os certificados. Portanto, uma aresta dirigida entre dois vértices K_u e K_v, $(K_u \rightarrow K_v)$, denota um certificado, assinado por u, amarrando K_w ao nó w. Um caminho conectando dois vértices K_u e K_v é representado por $(K_u \rightsquigarrow K_v)$. Cada nó u mantém um repositório local de certificados atualizados, L_u, e um outro repositório local de certificados não atualizados, L_u^N [7]. O repositório local de certificados não atualizados contém os certificados que expiraram e foram considerados revogados.

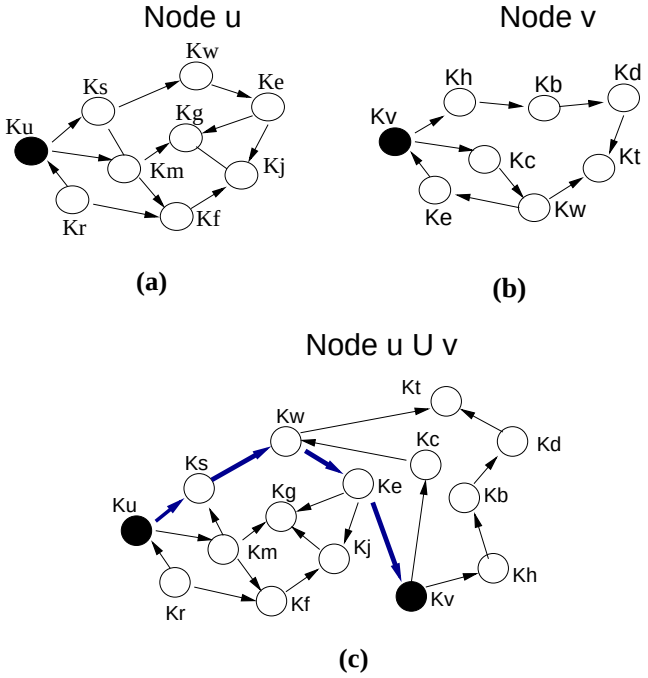


Figura 2.1: Repositórios locais de certificados dos nós u e v respectivamente *(a)* e *(b)* e e uma cadeia de certificado *(c)* [12].

Quando o nó u quer autenticar a chave pública K_v do nó v, ele primeiramente tenta achar um caminho do vértice K_u para o vértice K_v em L_u. Se $\exists(K_u \rightsquigarrow K_v) \in L_u$, nó u autentica nó v. Se $\neg\exists(K_u \rightsquigarrow K_v) \in L_u$, o nó u une L_u com L_v, $L_1 = L_u \cup L_v$, e tenta achar $(K_u \rightsquigarrow K_v) \in L_1$. Se tal caminho existe, a autenticação é realizada com sucesso. Se $\neg\exists(K_u \rightsquigarrow Kv) \in L_1$, então o nó u cria $L_2 = L_u \cup L_u^N$ e tenta encontrar $(K_u \rightsquigarrow K_v) \in L_2$. Se $\exists(K_u \rightsquigarrow K_v) \in L_2$, o nó u precisa validar todos os certificados expirados antes de usá-los. Se $\neg\exists(K_u \rightsquigarrow K_v) \in L_2$, então o nó u não consegue autenticar K_v.

O caminho encontrado nos repositórios é uma cadeia de certificados. As cadeias de certificados representam a confiança entre os nós e são chamadas de cadeias de confiança.

A Figura 2.1 mostra a união de dois repositórios locais de certificados para a construção de uma cadeia de certificados entre dois nós.

Note que cadeias de confianças são consideradas autenticações fracas, pois suas autenticações são transitivas. Por exemplo, se o nó a confia no nó b, e o nó b confia no nó c, então o nó a também confia no nó c. Infelizmente, garantir uma confiança transitiva válida entre mais de dois nós em uma cadeia é muito difícil [11]. Por tal motivo, se qualquer nó da cadeia for comprometido, todos os outros nós pertencentes à cadeia podem obter uma autenticação falsa.

O uso de cadeias de certificados torna o *PGP-Like* altamente vulnerável a ataques do tipo *Sybil*, como mostrado nas Figuras 2.2 e 2.3 [41]. A Figura 2.2 mostra o percentual de nós que contém identidades falsas em seus repositórios locais por tempo e para 5%, 10% e 20% de atacantes na rede. Com o passar do tempo, mais unidades falsas são encontradas nos repositórios locais, e maior a probabilidade de serem usadas em uma cadeia de certificados.

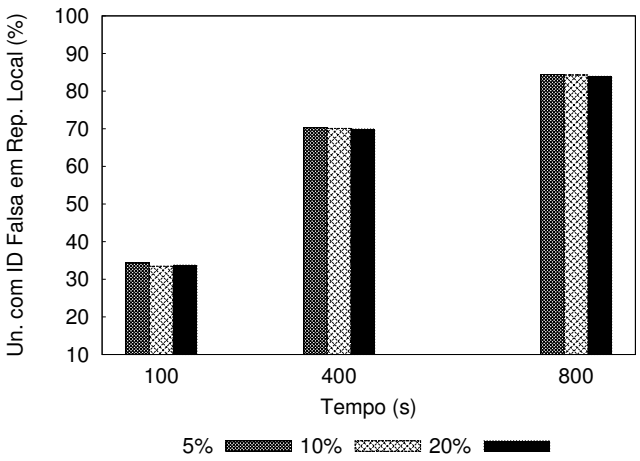


Figura 2.2: Confiança nas Identidades Falsas [41].

A Figura 2.3 mostra que as identidades falsas conseguem ser autenticadas por nós verdadeiros. Note que um nó atacante x pode criar uma identidade falsa m e emitir certificados amarrando K_m a m. Todos os nós que confiarem em x também confiarão em m. Portanto, se o nó x mantiver um comportamento correto durante um tempo considerável, várias unidades provavelmente irão confiar nele e a identidade falsa irá se

espalhar pela rede devido ao mecanismo de troca de certificados.

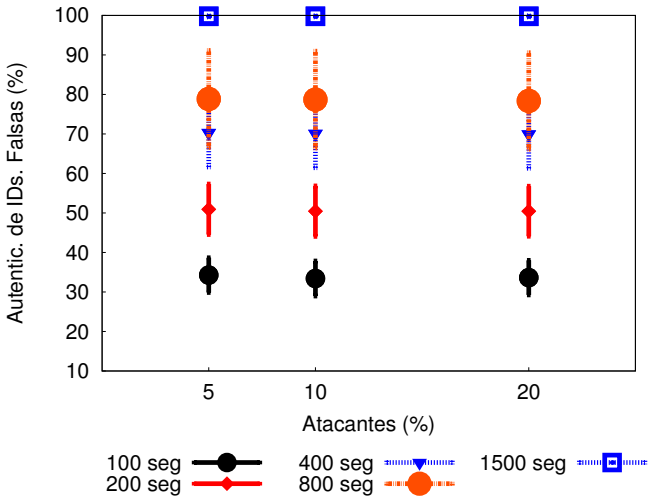


Figura 2.3: Certificados Falsos nos Repositórios Locais [41].

2.2 Group-based Key Management

Em [34] é proposto um esquema de gerenciamento de chaves baseado em grupos, referenciado neste trabalho por *Group-based Key Management* (GKM). No GKM, usuários formam pequenos grupos, nos quais todos os nós tem o mesmo papel sem a necessidade de haver um líder. Estes grupos são formados com base no relacionamento dos usuários que formam uma rede de grupos. Essa rede é usada para realizar todas as operações de gerenciamento de chaves e os nós precisam ser um membro de um grupo para fazer parte do sistema.

Cada nó i cria seu par de chaves públicas e privadas. Então, é necessário encontrar $m-1$ nós confiáveis e, com esses nós, formar um grupo. Os nós de um dado grupo trocam as suas chaves públicas entre si usando um canal seguro. Inicialmente os nós que formam o grupo devem gerar o par de chaves públicas e privadas do grupo. Então, eles geram juntos um par de chaves para aquele grupo. Posteriormente, cada nó de um dado grupo emite certificados para os outros $m-1$ nós. Estes certificados são assinados com a chave privada do grupo e armazenados. Então, no fim dessa fase todos os nós do grupo irão possuir certificados de todos os outros nós do grupo. Este tipo de certificado é chamado de *certificado de nó*. Como a chave pública de um grupo também precisa ser autenticada, os

grupos podem emitir certificados entre si associando uma dada chave pública a identidade de um grupo. Então, membros de um dado grupo IG_w podem emitir certificados para um grupo IG_z se confiarem no mesmo. Este tipo de certificado é chamado *certificado de grupo*.

Cada nó possuí quatro repositórios locais de certificados: dois para os grupos e nós atualizados e dois para os grupos e nós não atualizados. Repositórios atualizados mantém certificados que ainda são válidos. Quando o tempo de vida de um certificado expira, os respectivos certificados são armazenados no repositório local de certificados não atualizados. Certificados não atualizados precisam ser validados reativamente antes de serem utilizados. Os nós periodicamente trocam seus certificados de grupos com seus vizinhos físicos. Com a troca periódica de certificados, os nós aumentam o número de certificados de grupos nos seus repositórios.

Quando um nó i quer autenticar um nó j certificado por um grupo IG_w, ele precisa usar a chave pública de grupo PK_w. Entretanto, antes de usar uma chave pública de grupo, o nó i precisa autenticá-la. A autenticação da chave pública do grupo IG_w é realizada através de uma cadeia de certificados de grupo. Então, para autenticar a chave pública PK_w, o nó i procura por pelo menos duas cadeias de certificados de grupo válidos entre o seu grupo e o grupo IG_w no seu repositório local de certificados atualizados L_i. Se $\exists(K_i \rightrightarrows K_j) \in L_i$, ele pode então validar a chave pública PK_w do grupo IG_w e, então, o nó i pode validar o nó j.

Se $\neg\exists(K_i \rightrightarrows K_j) \in L_i$, i combina seu repositório de grupos atualizado com o do nó j. Então, o nó i tenta achar, novamente, ao menos duas cadeias de certificados de grupo. Se conseguir achar cadeias válidas, i poderá ser validar a chave pública PK_w do grupo IG_w e, então, poderá ser validado o certificado do nó j. Se mesmo após a junção dos repositórios, o nó i não achar uma cadeia de certificados válida, o nó i tenta encontrá-la na união dos repositórios atualizados e não atualizados. Caso a cadeia de certificado seja encontrada, o nó i precisa validar todos os certificados não atualizados. Caso a cadeia de certificados não seja encontrada, o nó i não será capaz de autenticar o certificado do nó j.

Os resultados apresentados em [12] mostram que o GKM consegue oferecer até 90% de segurança contra ataques do tipo *Sybil*. Nesses resultados não há distinção do tipo de ataque *Sybil*. Os resultados são os mesmos independente se os atacantes criam identidades falsas ou se os mesmos personificam outras unidades existentes na rede. A Figura 2.4 mostra a quantidade de autenticações realizadas que não contenham um nó comprometido. A efetividade do GKM só é afetada com 40% atacantes. Nesse caso, a métrica é aproximadamente 40% com o tamanho de grupo $G = 3$, enquanto que a métrica se mantém com 90% com $m = 6$. Note que os cenários de pior caso são aqueles com os menores grupos, pois os atacantes podem facilmente participar de grupos pequenos.

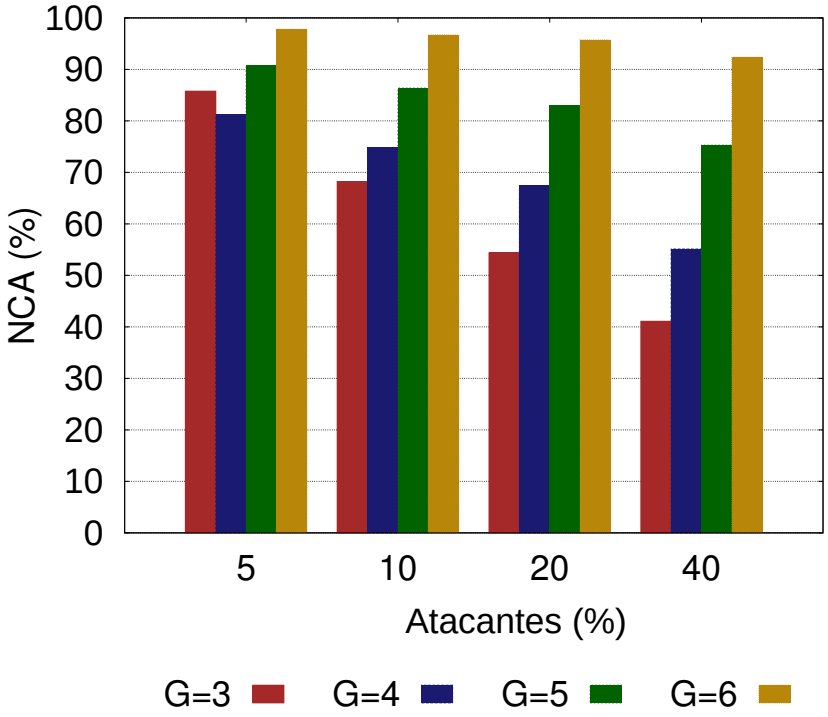


Figura 2.4: GKM: Autenticações realizadas através de nós não comprometidos.

CAPÍTULO 3

VIRTUAL KEY MANAGEMENT

Neste capítulo é apresentado um novo sistema de gerenciamento de chaves totalmente distribuído e auto-organizado. Esse novo esquema de gerenciamento de chaves utiliza uma estrutura virtual para definir a confiança entre os nós. Esse esquema pode ser utilizado por qualquer serviço que necessite de um esquema de gerenciamento de chaves na rede.

Nas próximas seções é apresentado um detalhamento do VKM. Na Seção 3.1 são apresentadas diferentes estruturas virtuais. Na Seção 3.2 é apresentado o funcionamento geral desse esquema de gerenciamento de chaves. Na Seção 3.3 é apresentado o VKM-RA que opera em modo reativo. Na Seção 3.4 é apresentado o VKM-PA que opera em modo pró-ativo. Finalmente, na Seção 3.5 é apresentada uma comparação entre os dois modos de operação do VKM.

3.1 Estruturas Virtuais

A estrutura virtual mais apropriada deve ser selecionada pelo usuário considerando propriedades como diâmetro, largura da bisseção e escalabilidade. O diâmetro de um grafo é a distância máxima de qualquer vértice do grafo. A largura da bisseção define quantos vértices são necessários remover para desconectar um grafo. Por fim a escalabilidade de um grafo implica na habilidade de manipular um grafo crescente de forma uniforme, estando o mesmo preparado para crescer.

Por exemplo, a estrutura virtual pode ser um *Ring of Rings*, um Hipercubo, um CCC ou um Torus. A Figura 3.1 caracteriza uma estrutura de *Ring of Rings*, com 3 anéis e 15 nós por anel. A Figura 3.2 um Hipecubo de dimensão $d = 3$. A Figura 3.3 caracteriza uma estrutura de Cubo de Ciclos de Conexos, com 3 nós por ciclo. A Figura 3.4 caracteriza uma estrutura 3D Torus.

A estrutura *Rings of Rings* (RoR) é baseada em congruências [44]. Assuma que

há dois inteiros, x e y, tal que, $x * y = n$, e seja s um inteiro tal que $1 < s \leq y$. O conjunto D é particionado em x anéis, chamados $D_0, D_1, ..., D_{x-1}$, onde, para cada $a \in [0, x)$, $D_a = \{i : a * y \leq i < (a + 1) * y\}$. O enlace (i, j) pertence a E se tanto $j \; mod \; y = (i + d) \; mod \; y$ para algum $1 \leq d < s$ ou $j = (i + y) \; mod \; n$. Uma característica da estrutura RoR é a redundância de caminhos virtuais, onde o grau é determinado por parâmetros x, y, e s.

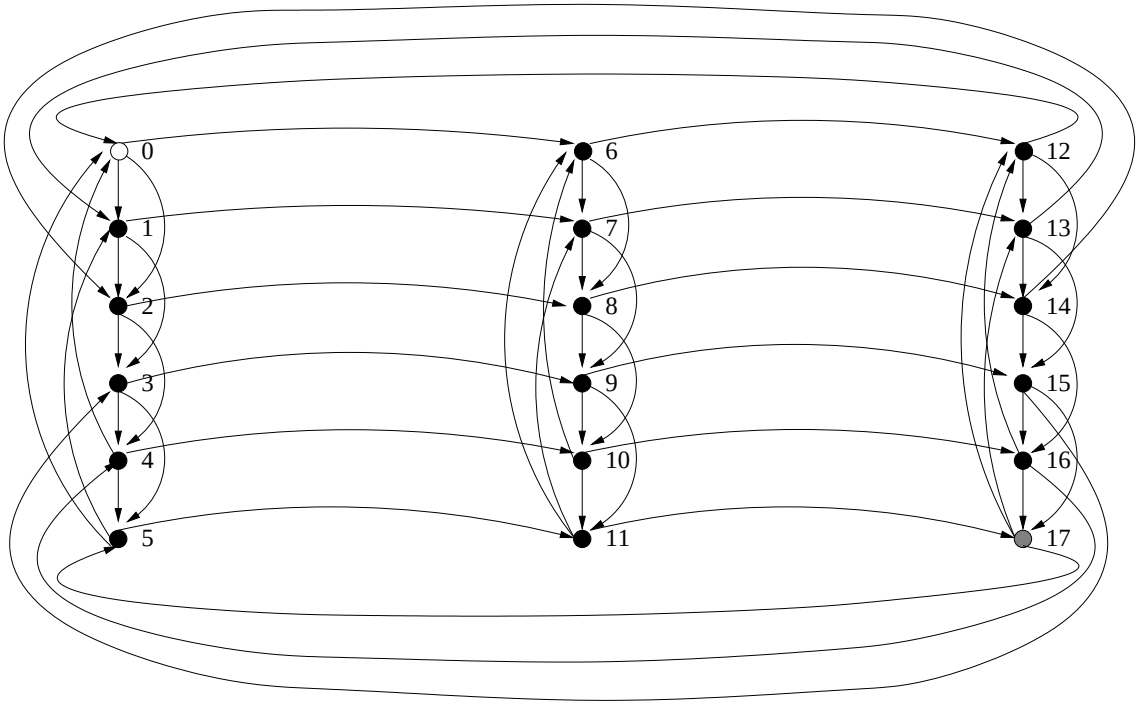


Figura 3.1: Estrutura virtual RoR [2].

Um *Hipercubo* de $k-$dimensões é um grafo regular que consiste em $n = 2^k$ vértices. O enlace entre os vértices é formado com base na distância $Hamming$. Uma vantagem no uso do Hipercubo é a distância máxima entre duas unidades ser igual a $log_2 n$. Entretanto, Hipercubos não são estruturas escaláveis [2].

Um *CCC* é uma estrutura formada por dois parâmetros, d e λ. d é a dimensão do Hipercubo que é formado pelos ciclos conexos e λ é o número de unidades por ciclo ($\lambda > d$). O número de ciclos da estrutura é 2^d e o total de vértices é $\lambda 2^d$. Um vértice será conectado a outros dois vértices dentro do ciclo e a uma unidade de outro ciclo. O grau dos vértices da estrutura é sempre 3. A vantagem de usar essa estrutura é que com um número menor de conexões é menor a quantidade de caminhos que cada unidade precisa

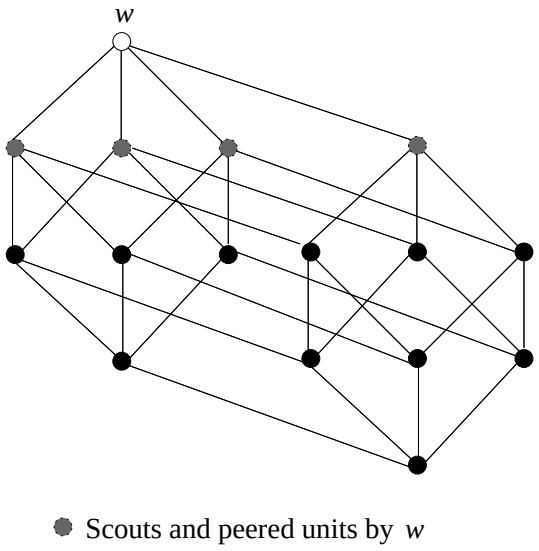


Figura 3.2: Estrutura virtual Hipercubo [2].

guardar pró-ativamente. Por outro lado, como o grau dos vértices é sempre 3, o CCC tem um número limitado de caminhos redundantes [2].

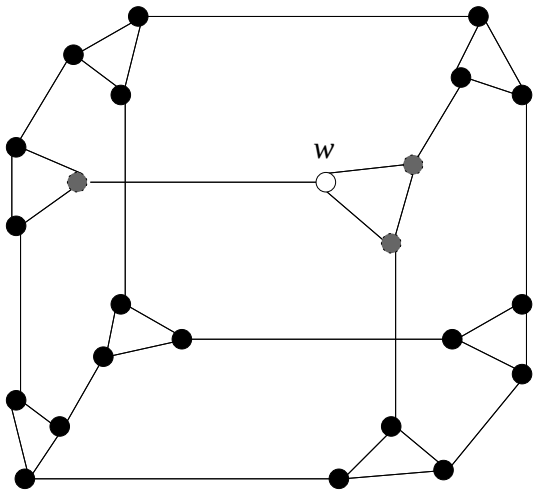


Figura 3.3: Estrutura virtual CCC [2].

O *3D Torus* de tamanho $n = l$xlxl é uma estrutura composta por n unidades distribuídas em l grades. O grau dos vértices é constante e igual a 6. Vértices no $3D$ $Torus$ podem ser endereçados pela tupla (x, y, z), sendo $x = 0, ..., l-1$, $y = 0, ..., l-1$ e $z = 0, ..., l-1$. Um vértice endereçado por (x, y, z) é conectado com vértices endereçados por $((x \pm 1)modl, y, z)$, $(x, (y \pm 1)modl, z)$ e $(x, y, (z \pm 1)modl)$. A distância máxima entre

unidades no 3D Torus é no máximo $\sqrt[3]{l}$ [2].

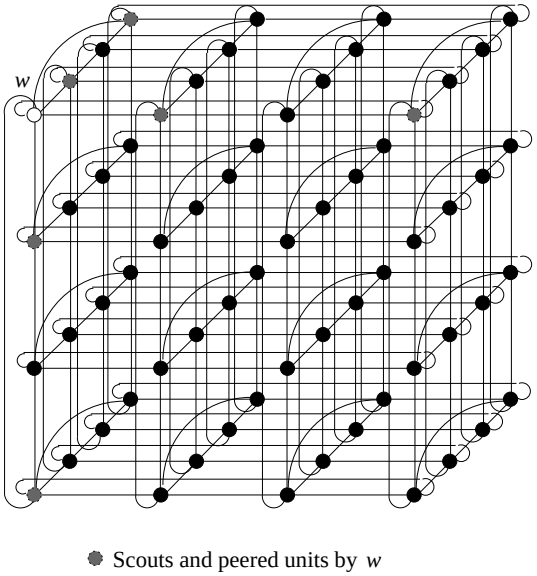


Figura 3.4: Estrutura virtual 3D Torus [2].

3.2 O *Virtual Key Management*

O esquema de gerenciamento de chaves *Virtual Key Management* (VKM)[18, 17] é baseado em certificados [7, 25] e, assim como o PGP-Like, é auto-organizável e segue os conceitos do PGP [48], no qual todo par de chaves pública e privada é criado pelos próprios nós da rede. O VKM usa uma *estrutura virtual* para indicar a relação de confiança entre os nós e a formação de cadeias de certificados. A estrutura virtual, que definirá a emissão de certificados na rede, é representada por um grafo dirigido $L = (D, E)$, o qual não está relacionado com a topologia atual da rede. O conjunto D representa os nós e o conjunto E representa os enlaces virtuais. Um enlace virtual $(u, v) \in E$ indica que o nó i emite um certificado associando K_j ao nó j. Note que o nó i precisa fazer este procedimento para cada nó que tenha uma conexão direcionada na estrutura virtual.

É importante mencionar que o VKM é independente da implementação do grafo da estrutura virtual. Entretanto, percebe-se que o grafo deve ser regular para garantir que o número de arestas seja o mesmo para todos os nós. Os resultados reportados nesse trabalho foram obtidos utilizando o grafo *Rings of Rings* (RoR), que será detalhado abaixo.

A Figura 3.5 exemplifica o grafo *Ring of Rings* (RoR) [2], com 45 nós, dividido em 3 anéis de 15 nós. Ainda na figura 3.5, cada nó tem uma conexão direta para outros cinco nós, significando que eles são responsáveis por emitir cinco certificados amarrando a chave desses nós. Por exemplo, o nó w é responsável por emitir certificados amarrando K_{w_1} à w_1, K_{w_2} à w_2, K_{w_3} à w_3, K_{w_4} à w_4 e K_{w_5} à w_5, e nós w'_1, w'_2, w'_3, w'_4 e w'_5 são responsáveis por emitir certificados amarrando K_w à w.

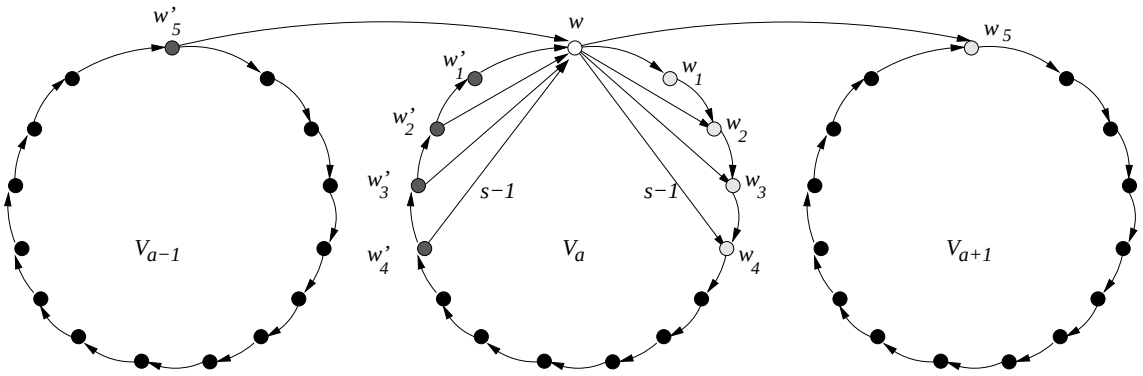


Figura 3.5: Estrutura virtual RoR com 3 anéis e 15 nos por anel

No VKM, cada nó i cria seu próprio par de chaves pública e privada, K_i e prK_i. Posteriormente, ele precisa emitir certificados seguindo a estrutura virtual. Um par de nós na estrutura virtual precisa trocar suas chaves através de um canal seguro como infravermelho, *smart cards* ou antes da formação da rede.

Todos os certificados são emitidos com um tempo de vida limitado T_j, e após T_j, o certificado é considerado expirado. Antes da expiração do certificado, o nó que o emitiu pode atualizar o certificado, emitindo uma nova versão com um T_j estendido. A revogação de certificados pode ser feita de uma maneira explícita ou implícita. Certificados revogados implicitamente são baseados no tempo T_j. Se um emissor de certificado não atualiza seu certificado após T_j, o certificado é considerado revogado. Na revogação explícita, o nó que o emitiu revoga o certificado se suspeita de mal comportamento do outro nó. A tarefa de detecção de mal comportamento dos nós da rede não faz parte do escopo deste trabalho.

Quando um certificado é emitido, o nó que emitiu o certificado o armazena em seu repositório local e envia esse certificado para o nó correspondente, que também armazena o certificado. Então, no início do tempo de vida da rede, os nós armazenam apenas

os certificados que eles emitiram e certificados que foram emitidos para ele. Como será demonstrado mais adiante, o uso da estrutura virtual torna o VKM muito flexível, podendo se comportar de uma maneira restrita e sendo completamente resistente a ataques de criação de identidades falsas e capaz de suportar ataques de personificação até um certo nível, ou podendo se comportar de maneira similar ao *PGP-Like*, apenas mudando alguns parâmetros. A principal diferença entre os comportamentos é a maneira como os nós autenticam as chaves públicas. Ambas as formas serão apresentadas nas próximas seções.

3.3 VKM com autenticação reativa

No modo de autenticação reativa, *VKM with Reative Authentication* (VKM-RA), cada nó mantém apenas seus certificados iniciais, ou seja, os certificados emitidos para ele e os certificados que ele emitiu. Se um nó i acredita que uma dada chave pública K_j pertence a um dado nó j, ele pode emitir um certificado amarrando K_j ao nó j, $(j, K_j)_{prK_i}$, onde prK_i é a chave privada do nó i. Este certificado é armazenado no repositório local de certificados de i e de j.

Quando um nó i quer autenticar a chave pública de um nó k, ele precisa achar um caminho virtual de i até k, uma cadeia de certificados, na estrutura virtual. Note que é possível achar vários caminhos virtuais de i até k, uma vez que a estrutura virtual é altamente redundante. Após escolher um caminho virtual, a origem precisa obter todos os certificados dos nós que compõem o caminho virtual para validá-lo ou seja, validar toda a cadeia de certificados. Cada certificado precisa ser requisitado ao nó que o emitiu.

Note que, todos os nós conhecem a mesma estrutura virtual, mas eles não mantêm informações atualizadas sobre os certificados, por exemplo, se eles foram revogados. Diferentemente do *PGP-Like*, uma unidade precisa manter apenas os certificados emitidos para ela e os certificados emitidos por ela, reduzindo, portanto, a memória necessária para armazenar os certificados. Entretanto, a origem precisa validar todos os certificados da cadeia de certificados e a autenticação é realizada da seguinte maneira:

1. o primeiro certificado pode ser diretamente verificado pelo nó i usando sua própria chave púbica, uma vez que foi ele que emitiu o certificado;

2. cada certificado remanescente pode ser verificado usando a chave pública contida no certificado anterior;

3. finalmente, o último certificado contém a chave pública do nó j.

Este comportamento garante que apenas certificados corretos e válidos sejam utilizados. Entretanto, como os nós precisam requisitar todos os certificados de uma cadeia antes de autenticar a chave pública, isto implica em uma latência ao fazer as autenticações. Por outro lado, o VKM-RA usa muito pouca memória para armazenamento local. Uma unidade precisa manter armazenado localmente apenas os certificados emitidos por ela, os certificados emitidos para ela e uma pequena função para criar e usar a estrutura virtual. Se a rede fizer uso de um protocolo de roteamento que utilize uma estrutura virtual, como o VRP [2] ou o VDV [40], o VKM pode usar a mesma estrutura virtual para autenticação das chaves.

3.4 VKM com autenticação pró-ativa

No VKM com autenticação pró-ativa, *VKM with Proactive Authentication* (VKM-PA), a fase inicial do VKM-PA é igual a do VKM-RA. Adicionalmente, cada nó realiza trocas de seu repositório com seus vizinhos físicos periodicamente. Por simplicidade e sem perda de generalidade, é assumido que todos os nós possuem o mesmo intervalo de troca T_{ex} e que o mesmo não é simétrico.

Cada nó i mantém um repositório local de certificados atualizados, L_i, e um outro repositório local de certificados não atualizados, L_i^N. O repositório local de certificados não atualizados contém os certificados que expiraram e foram considerados revogados.

Quando o nó i quer autenticar a chave pública K_j do nó j, ele primeiramente tenta achar um caminho do vértice K_i para o vértice K_j em L_i. Em outras palavras, quando um nó i quer autenticar uma chave pública K_j do nó j, ele tenta achar uma cadeia de certificados no repositório local. Se $\exists(K_i \rightsquigarrow K_j) \in L_i$, nó i autentica nó j. Se

$\neg\exists(K_i \rightsquigarrow K_j) \in L_i$, nó i une L_i com L_j, $L_1 = L_i \cup L_j$, e tenta achar $(K_i \rightsquigarrow K_j) \in L_1$. Se tal caminho existe, a autenticação é realizada com sucesso. Se $\neg\exists(K_i \rightsquigarrow Kv) \in L_1$, então o nó i cria $L_2 = L_i \cup L_i^N$ e tenta encontrar $(K_i \rightsquigarrow K_j) \in L_2$. Se $\exists(K_i \rightsquigarrow K_j) \in L_2$, o nó i precisa validar todos os certificados expirados antes de usá-los. Se $\neg\exists(K_i \rightsquigarrow K_j) \in L_2$, então o nó i invoca o uso do VKM-RA. Esta característica faz a autenticação do VKM mais eficiente que a do *PGP-Like*, uma vez que é possível atingir todos os nós da rede usando VKM-RA.

Considerando a troca de certificados armazenados nos repositórios locais, o VKM-PA tem um comportamento similar ao *PGP-Like*. Por outro lado, no *PGP-Like* os nós dependem de um tempo de convergência após a inicialização da rede, que nada mais é do que o tempo para popular os repositórios locais de certificados de boa parte dos nós. Se o tempo de funcionamento da rede é maior do que o tempo de convergência, esse não é um problema a ser observado. Porém, quando o tempo de funcionamento da rede é menor do que o tempo de convergência necessário para os nós terem um grafo conexo em seus repositórios locais de certificados, o *PGP-Like* pode não ser capaz de autenticar um nó na rede. Com isso, o VKM expande a capacidade de autenticação dos nós, mesmo operando no modo pró-ativo.

3.5 Comparando o VKM-RA e o VKM-PA

Como no VKM-RA os nós não realizam trocas de seus repositórios de certificados, a sobrecarga das trocas de repositórios é eliminada, mas isto inclui um atraso, necessário para validar cada certificado da cadeia. Este atraso depende do protocolo de roteamento usado pela rede e do tamanho das cadeias de certificados. São necessárias duas mensagens para cada nó da cadeia de certificados. Por exemplo, na Figura 3.6 o nó i precisa enviar uma mensagem para o nó k para validar $(l, K_l)_{prK_k}$ e uma mensagem para o nó l para validar $(j, K_j)_{prK_l}$. Após receber ambas as respostas, i pode autenticar K_j.

Figura 3.6: Cadeia de certificados

Por outro lado, o VKM-PA pode eliminar o atraso na autenticação se uma cadeia de certificados válida e atualizada for encontrada nos repositórios locais. Senão, é necessário invocar o VKM-RA. Portanto, o atraso do VKM-PA pode ser menor ou igual ao do VKM-RA dependendo da completude e validade dos repositórios locais de certificados. O VKM-PA possui uma sobrecarga para realizar as trocas de certificados. Outro problema observado é a quantidade de memória de armazenamento necessária para os nós. Em MANETs, os nós muitas vezes dispõem de recursos de hardware limitados, e portanto, devem ser administrados racionalmente. Enquanto no VKM-RA os nós só mantêm cópias de alguns certificados, no VKM-PA e no *PGP-Like*, os nós irão eventualmente manter cópias de todos os certificados da rede. Quando o espaço de armazenamento das unidades da rede for um problema, os nós podem configurar o modo de funcionamento para o VKM-RA, evitando a troca de certificados e economizando espaço de armazenamento para outras aplicações. Por fim, outra importante característica do VKM é a habilidade de alterar dinamicamente entre os dois modelos apresentados, sem reinicialização da rede ou mesmo sem qualquer reconfiguração.

CAPÍTULO 4

RESULTADOS DOS ESQUEMAS DE GERENCIAMENTO DE CHAVES

Neste capítulo serão apresentadas as avaliações de simulações para verificar a efetividade do VKM. O VKM foi comparado com o GKM em ambientes com ataques de do tipo *Sybil* (ataques de personificação e ataques de criação de identidades falsas). O VKM também foi comparado com o *PGP-Like* em ambientes sem ataques e em ambientes na presença de ataques de falta de cooperação e na presença de ataques de personificação.

O simulador *Network Simulator* 2 (NS-2) [35], em sua versão 30, foi usado para verificar a efetividade do VKM quando submetido a ataques de personificação e ataques de criação de identidades falsas. A versão 30 foi escolhida nessa comparação pois os resultados aqui apresentados foram comparados com os resultados do PGP-Like e do GKM apresentados em [12], também implementados na versão 30. Os parâmetros usados nas simulações estão na Tabela 4.1 e foram escolhidos por serem os mesmos utilizados na comparação do PGP-Like e do GKM [12]. Os resultados são médias de trinta e cinco simulações com intervalo de confiança de 95%. A estrutura virtual escolhida foi o RoR com quatro anéis e vinte e cinco nós por anel. Cada nó emite cinco certificados e tem cinco certificados emitidos para ele. O RoR foi escolhido por ser uma estrutura de fácil adaptação, onde é possível alterar a conectividade da estrutura virtual sem alterar o diâmetro do grafo. Dessa forma a estrutura se apresenta bastante flexível para simular diferentes cenários.

4.1 Avaliação do VKM-RA

No VKM-RA, ataques onde nós maliciosos criam falsas identidades para obter alterar o correto funcionamento da rede não seriam eficazes. Os nós maliciosos devem conhecer a lógica da estrutura virtual para participar de aplicações na rede. Portanto, se um atacante utilizar uma identidade falsa que não faça parte da estrutura virtual ou mesmo se ele criar

Tabela 4.1: Cenários das simulações do VKM avaliado sob ataques

Parâmetros	**Valor**
Dimensão da rede	1000 x 1000 metros
Alcance da transmissão	120 metros
Nós	100
Modelo de mobilidade	*random waypoint*
Velocidade máxima	20 m/s
Tempo de pausa máximo	20 segundos
Tempo de Simulação	1500 segundos
Modelo de Propagação	*two-ray ground reflection*
Protocolo de Acesso ao Meio	IEEE 802.11

identidades falsas para atacar a rede, o ataque será completamente sem sentido, uma vez que essa identidade falsa nem chegará a ser autenticada.

Para avaliar o VKM-RA sob o ataque de personificação, uma nova métrica é proposta: *Chains with Sybil Nodes* (CSN) - Cadeias com Nós Personificados e *No Compromised Authentications* - (NCA) - Autenticações não comprometidas. Considerando todos os caminhos virtuais possíveis (VP), a métrica CSN é a porção de caminhos virtuais, ou seja, cadeias de certificados, que contêm ao menos uma identidade falsa (m). CSN pode ser definido como:

$$CSN = \frac{\sum CSN_i}{VP|} \quad \forall\, i \in \{VP\} \qquad \text{onde} \tag{4.1}$$

$$CSN_i = \begin{cases} 1 & \text{se} \quad \exists m \in (K_i \rightsquigarrow K_j) \\ 0 & \text{caso contrário} \end{cases} \tag{4.2}$$

Considerando todos os caminhos virtuais possíveis (VP), a métrica NCA é a porção de caminhos virtuais, ou cadeias de certificados, que contém ao menos uma identidade falsa (m). NCA pode ser definida como:

$$NCA = \frac{\sum CSN_i}{|VP|} \quad \forall\, i \in \{VP\} \qquad \text{onde} \tag{4.3}$$

$$NCA_i = \begin{cases} 0 & \text{se} \quad \exists m \in (K_i \rightsquigarrow K_j) \\ 1 & \text{otherwise} \end{cases} \tag{4.4}$$

Os cenários das simulações consideram 5%, 10%, 20% e 40% de nós atacantes. Sendo S o número de certificados emitidos para e por cada nó, os cenários também consideram $S = 5$, $S = 10$, $S = 15$, e $S = 20$; e $S = 3$, $S = 4$, $S = 5$ e $S = 6$ utilizando o VKM sobre as métricas CSN e NCA, respectivamente; e $G = 3$, $G = 4$, $G = 5$ e $G = 6$ sendo o tamanho dos grupos do GKM sobre a métrica NCA. Como mostrado na Figura 4.1, mesmo com 20% de atacantes na rede, o VKM é capaz de autenticar mais de 40% de cadeias de certificados não comprometidas. Na presença de 5% de atacantes, o VKM-RA é capaz de autenticar corretamente aproximadamente 80% das cadeias de certificados, enquanto *PGP-Like* é completamente vulnerável, mesmo com apenas 5% de atacantes [41].

O VKM-RA pode tolerar ataques de personificação melhor que o *PGP-Like* devido à estrutura virtual, uma vez que a mesma é altamente redundante e estabelece várias cadeias fixas para autenticação. Se o número de nós comprometidos é pequeno, é possível evitar os nós comprometidos apenas escolhendo a cadeia de certificado de maneira aleatória.

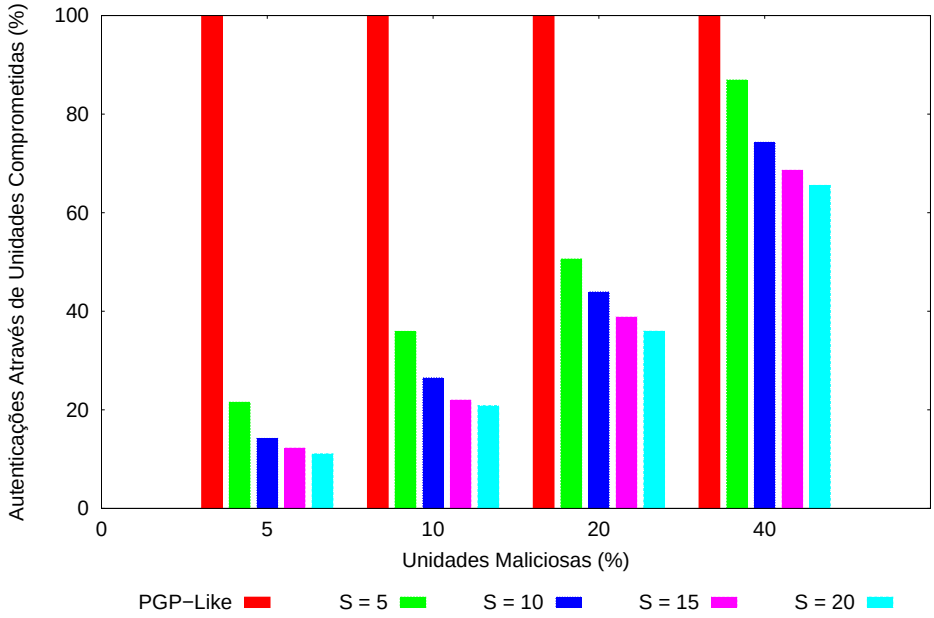
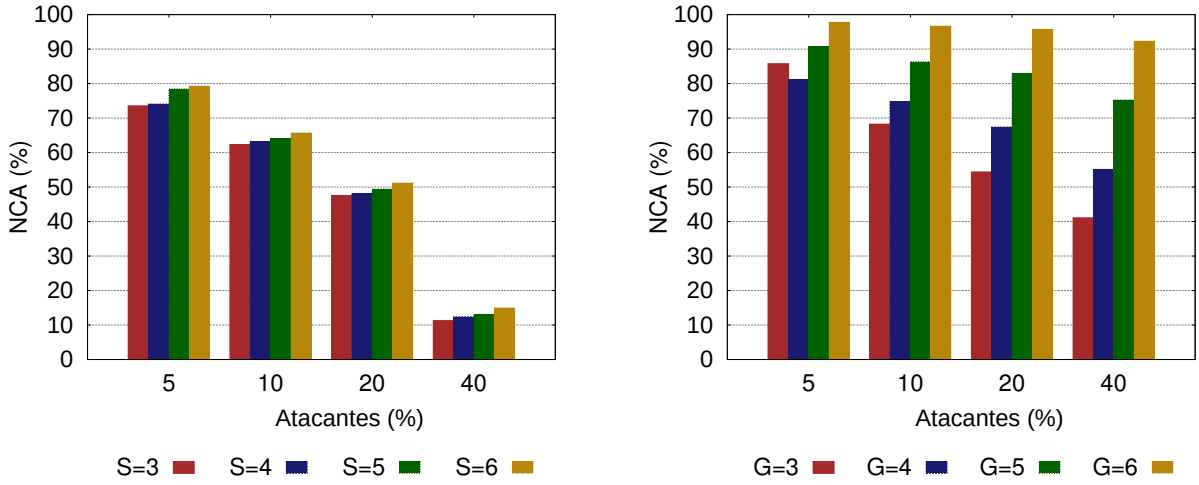


Figura 4.1: Autenticações feitas através de nós comprometidos.

A Figura 4.1 também mostra que com 40% de atacantes e cinco certificados emitidos por nó, a possibilidade de escolher uma cadeia de certificados comprometida chega a mais de 80%. Entretanto, esse número é reduzido para aproximadamente 60% se o número

de certificados emitidos for aumentado para 20. Demonstrando, portanto, que com um aumento na conectividade da estrutura virtual é possível reduzir ainda mais os efeitos de ataques de personificação.

Os resultados mostram que o VKM-RA é efetivo contra ataques de personificação desde que a quantidade de atacantes seja menor do que 60% e aleatória. Se a quantidade de atacantes for maior do que 60% ou se os atacantes organizarem um ataque cooperativo a um nó (ou a uma região) da estrutura virtual, separando tal nó (ou região) do resto da rede, o ataque será eficaz. Entretanto, isso somente é possível se os atacantes conhecerem a estrutura virtual e ainda realizarem um ataque a vários nós de maneira simultânea, desconectando a estrutura virtual.



(a) VKM (b) GKM

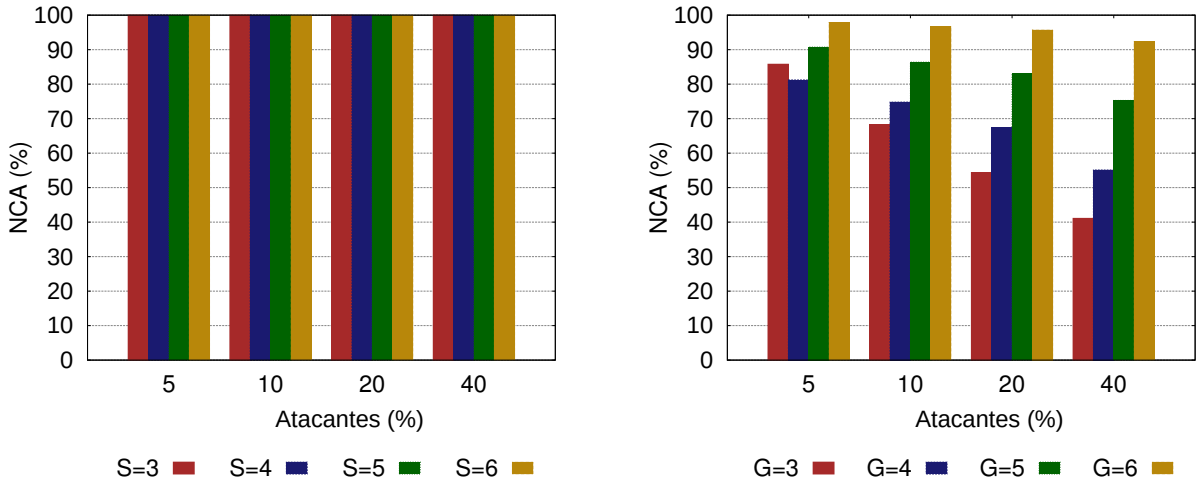
Figura 4.2: Autenticações realizadas através de nós não comprometidos (personificados) - VKM x GKM

A Figura 4.2 mostra o impacto do ataque de personificação sobre o VKM e o GKM usando a métrica NCA. No GKM é necessário achar dois certificados de grupo disjuntos para autenticar uma chave pública. Isso implica que diversas identidades falsas no sistema não poderão ser autenticadas. Por outro lado, no VKM os nós confiam completamente na estrutura virtual. Se um nó que conseguir acesso a estrutura virtual estiver comprometido, todas as autenticações que passarem por este nó estarão comprometidas.

Aumentando a quantidade de nós personificados diretamente afeta a estrutura virtual e, consequentemente, mais unidades comprometidas participarão de autenticações. As au-

tenticações no VKM seguem seleções de caminhos na estrutura virtual de forma aleatória, e aumentando a conectividade da estrutura virtual (S) reduz o impacto do ataque de nós personificados. Em um cenário com 40% de nós atacante e $S = 6$, o valor de NCA é aproximadamente 13%, enquanto que com 5% de atacantes, NCA é aproximadamente 78%. É importante mencionar que isso ocorreria apenas em um cenário onde a unidade personificada fosse capaz de participar da estrutura virtual, podendo enfim, participar das autenticações realizadas na rede. A participação de apenas nós não comprometidos no VKM cai acima de 10% de atacantes.

No GKM, a efetividade só é afetada com 40% atacantes. Neste caso, NCA é aproximadamente 40% com o tamanho de grupo $G = 3$, enquanto que a métrica se mantém com 90% com $G = 6$. Note que os cenários de pior caso são aqueles com os menores grupos, pois os atacantes podem facilmente participar de grupos pequenos. Em comparação, o impacto em relação a quantidade de certificados emitidos por cada nó não é tão diferente em proporção ao tamanho do grupo, sendo que um atacante que tenha acesso a estrutura virtual terá mais chance de ser utilizado em autenticações da rede quando poucos certificados são emitidos por cada nó.



(a) VKM

(b) GKM

Figura 4.3: Autenticações realizadas através de nós não comprometidos (identidades falsas) - VKM x GKM

s

A Figura 4.3 mostra o impacto do ataque de criação de identidades falsas sobre o VKM

e o GKM usando a métrica NCA. No GKM os resultados para ataques de personificação são os mesmos para ataques de criação de identidades falsas. Isso ocorre pois não há distinção entre as mesmas no momento de formação dos grupos. Por outro lado, no VKM os nós confiam completamente na estrutura virtual. Se um nó criar identidades falsas que não façam parte da estrutura virtual, elas não poderão ser autenticadas ou fazer parte da autenticação de outras unidades.

Como no VKM a criação de identidades falsas não afeta as autenticações realizadas pelos nós que fazem parte da estrutura virtual, a efetividade é de 100%, independente do número de atacantes. Somente se fosse possível convencer todos os nós da rede a remodelar a estrutura virtual um atacante obteria sucesso ao conseguir que uma identidade falsa participe nas autenticações da rede. Porém, isso seria inviável pois dificilmente uma unidade qualquer obteria acesso a nova estrutura virtual. A tarefa de decidir quais nós farão parte da estrutura virtual após a sua remodelagem não foi abordada neste trabalho.

Nos resultados apresentados é possível concluir que o GKM é menos afetado por ataques de personificação. Isto ocorre devido a necessidade de duas cadeias de certificados distintas serem necessárias na rede virtual de grupos. É importante mencionar que é possível aumentar a a resistência do VKM ao aumentar a conectividade da estrutura virtual, ou até mesmo solicitar que cada origem ache dois caminhos disjuntos dentro da estrutura virtual (apesar de não ter sido avaliado neste trabalho o impacto de tal modificação). Por outro lado o VKM é virtualmente 100 % resistente a ataques de criação de identidade falsas, enquanto o GKM mantém os mesmos resultados apresentados no ataque de personificação. Isso ocorre por que no VKM os nós que fazem parte da estrutura virtual são diferenciados em relação a demais nós da rede. No GKM não há distinção entre os nós que fazem parte dos grupos.

Ambos os esquemas podem utilizar mecanismos de detecção de mal comportamento ou de reputação, apesar de nenhum dos dois esquemas ainda ter apresentado estudos ou resultados com tais mecanismos. A Tabela 4.2 apresenta as principais características entre os esquemas de gerenciamento de chaves VKM e GKM. No VKM, a estrutura virtual é estática e conhecida por todos os nós. Portanto, este esquema não permite a entrada de

novos nós na rede facilmente. Esta característica pode não ser desejável em ambientes MANETs, onde a entrada de novos nós na rede é muito comum. Entretanto, há situações onde é mais fácil prever os nós que farão parte da rede como comunicações privadas, onde os participantes são conhecidos desde o começo da comunicação. Adicionalmente, o VKM permite que os nós saiam do sistema a qualquer momento, afetando apenas a quantidade de cadeias de certificados válidas disponível na estrutura virtual.

Tabela 4.2: Principais Características do GKM e do VKM

	Group-based	**VKM**	
		RA	PA
Rede Virtual	Dinâmica	Fixa, mas pode ser redefinida	
Sobrecarga	Troca de certificados + validação reativa de certificados	Validação reativa de certificados	Troca de certificados + validação reativa de certificados
Escalabilidade	Sim	Não	
Tamanho dos Repositórios Locais	Todos os certificados de grupo + todos os certificados do nós	Certificados emitidos por/para os nós	Todos os certificados dos nós
Validação Reativa de Certificados	Sim	Sim	
Atacantes externos	Sim, se respeitada as regras de formações dos grupos	Não	
Alcançabilidade dos Nós	Após o período de convergência	Desde a formação da rede	
Autenticações	Cadeias de certificados de grupo	Cadeias de certificados na estrutura virtual	Cadeias de certificados nos repositórios locais + cadeias de certificados na estrutura virtual

No VKM-PA, repositórios locais de certificados são formados através de um mecanismo de troca de certificados. Todas as autenticações de chaves são feitas ou usando os repositórios locais de certificados ou invocando o VKM-RA. No VKM-RA, autenticações de chaves são feitas de maneira reativa e as cadeias de certificados devem ser validadas. No GKM há uma maior sobrecarga de comunicação para manter os grupos. Ainda, um certificado de grupo precisa ser validado em conjunção por, ao menos, t nós que pertençam ao grupo certificado. Portanto, para validar uma cadeia de certificados de grupo com tamanho k, seria necessário $k * t$ mensagens.

Sem divergências sobre demais estudos, é possível enfatizar as vantagens no uso de virtualização quando aplicada em esquemas de gerenciamento de chaves sem considerar

diferentes opções de parâmetros. Portanto, apesar de haver uma queda na efetividade contra ataques de personificação (devido ao não uso de mais de um caminho de certificados para autenticar uma unidade), o uso de virtualização oferece 100% de segurança contra ataques de criação de identidades falsas.

4.2 Avaliação do VKM-PA

O VKM-PA não foi avaliado sob ataques de personificação, pois, igualmente ao *PGP-Like*, ele é completamente vulnerável devido às trocas dos repositórios locais de certificados dos nós. Entretanto, o VKM-PA é submetido a ataques de falta de cooperação pois espera-se que o mesmo seja no mínimo tão eficiente contra esse tipo de ataque.

As simulações consideram 5%, 20%, 40%, 60% e 80% de nós egoístas. Tais nós realizam todas as funções básicas da rede, emitindo, inclusive, certificados. Entretanto, eles não cooperam com o mecanismo de troca de certificados, não solicitam nem aceitam certificados de seus vizinhos. As simulações consideraram um tempo de vida de 1500 segundos e os certificados foram emitidos na inicialização da rede. O intervalo de tempo para realizar a troca de certificados é de 60 segundos.

Seguindo os resultados apresentados em [7] e [41] para avaliar o *PGP-Like*, duas métricas são usadas nas avaliações: CE (*Certificate Exchange Convergence*) e UR (*User Reacheability*). CE e UR medem a completude dos repositórios locais de certificados e a utilidade das trocas de certificados, respectivamente. De acordo com [41], CE e UR podem ser definidos como:

$$CE(t) = \frac{\sum CE\_i(t)}{|S|} \quad \forall i \in \{S\} \quad \text{onde} \tag{4.5}$$

$$CE\_i = \frac{\sum |(K_a \rightsquigarrow K_b) G_i^N \ \cup \ G_i|}{\sum |(K_x \rightsquigarrow K_y) \in G|} \quad \forall \ a, b, x, y \in \{S\} \tag{4.6}$$

$$UR(t) = \frac{\sum UR\_i(t)}{|S|} \quad \forall i \in \{S\} \quad \text{onde} \tag{4.7}$$

$$UR\_i = \frac{\sum |(K_i \rightsquigarrow K_a) \in G_i^N \ \cup \ G_i|}{\sum |(K_i \rightsquigarrow K_x) \in G|} \quad \forall a, x \in \{S\} \tag{4.8}$$

As Figuras 4.4 e 4.6 ilustram o comportamento do VKM-PA em ambientes sem ataques e sob ataques de falta de cooperação. Em ambos, os resultados do VKM-PA são comparáveis aos do *PGP-Like*. Como esperado, aumentando o número de atacantes, o valor de CE diminui (Figura 4.4).

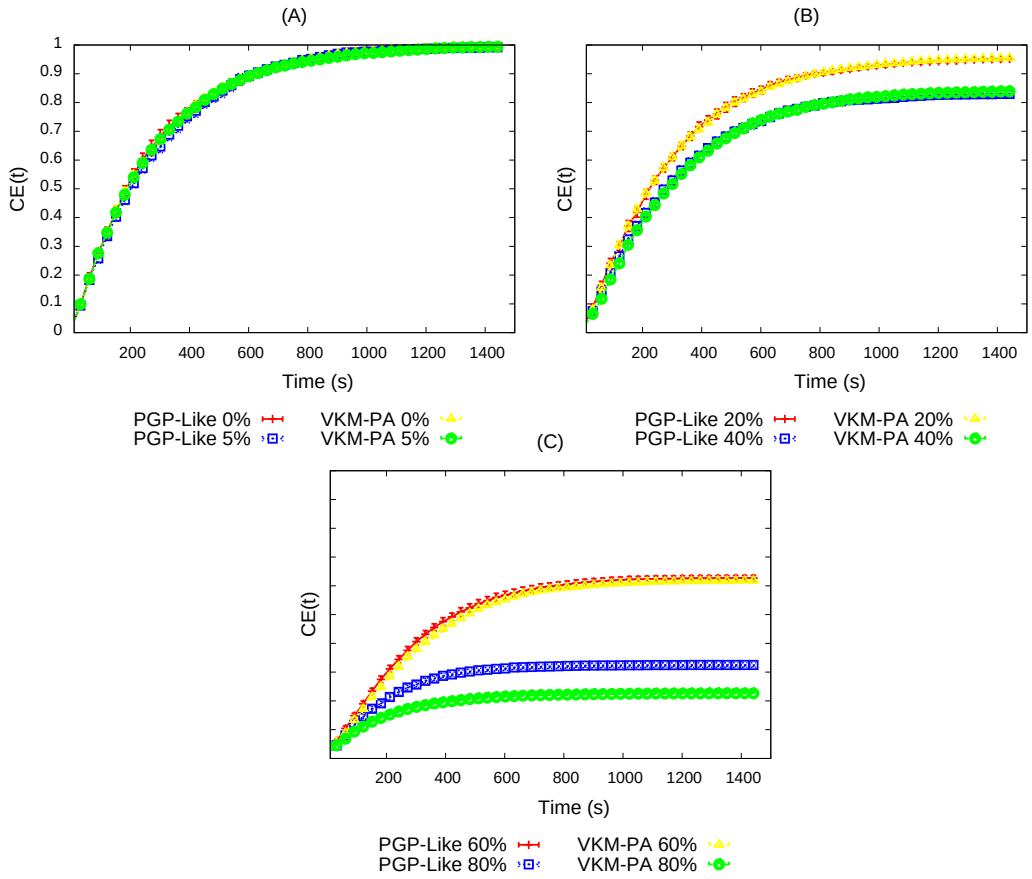


Figura 4.4: Convergência de trocas de certificados sob ataques de falta de cooperação.

Observe que em cenários sem atacantes ou com 5% a 60% de atacantes, o VKM-PA apresenta um comportamento idêntico ao *PGP-Like*. Apenas em cenários com 80% de atacantes, o desempenho do VKM-PA é cerca de 8% menor que a do *PGP-Like*. Isto se deve à desconexão da estrutura virtual como mostrado na Figura 4.5.

Os resultados do UR para o VKM-PA com até 60% de nó egoístas são também muito similares aos do *PGP-Like* (Figura 4.6). O UR é quase 100%, mesmo na presença de 60% de nós egoístas. Entretanto, quando o número de atacantes sobe para 80%, o desempenho

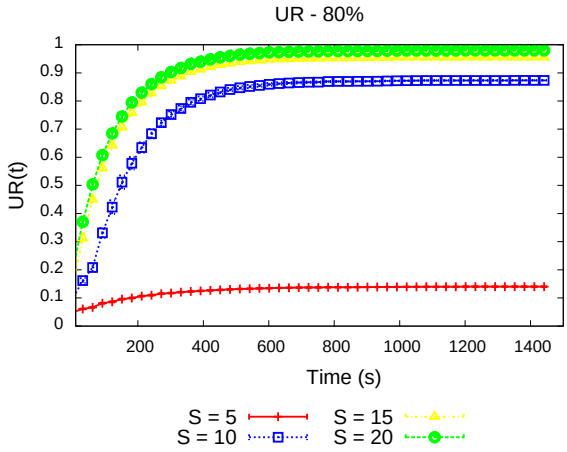


Figura 4.5: *User Reacheability* sob 80% de nós egoístas.

do VKM-PA diminui drasticamente, ficando abaixo de 10%. Novamente, neste caso, a estrutura virtual fica desconexa e o VKM-PA não consegue encontrar cadeias de certificados. Uma maneira de superar este problema é aumentando a conectividade da estrutura

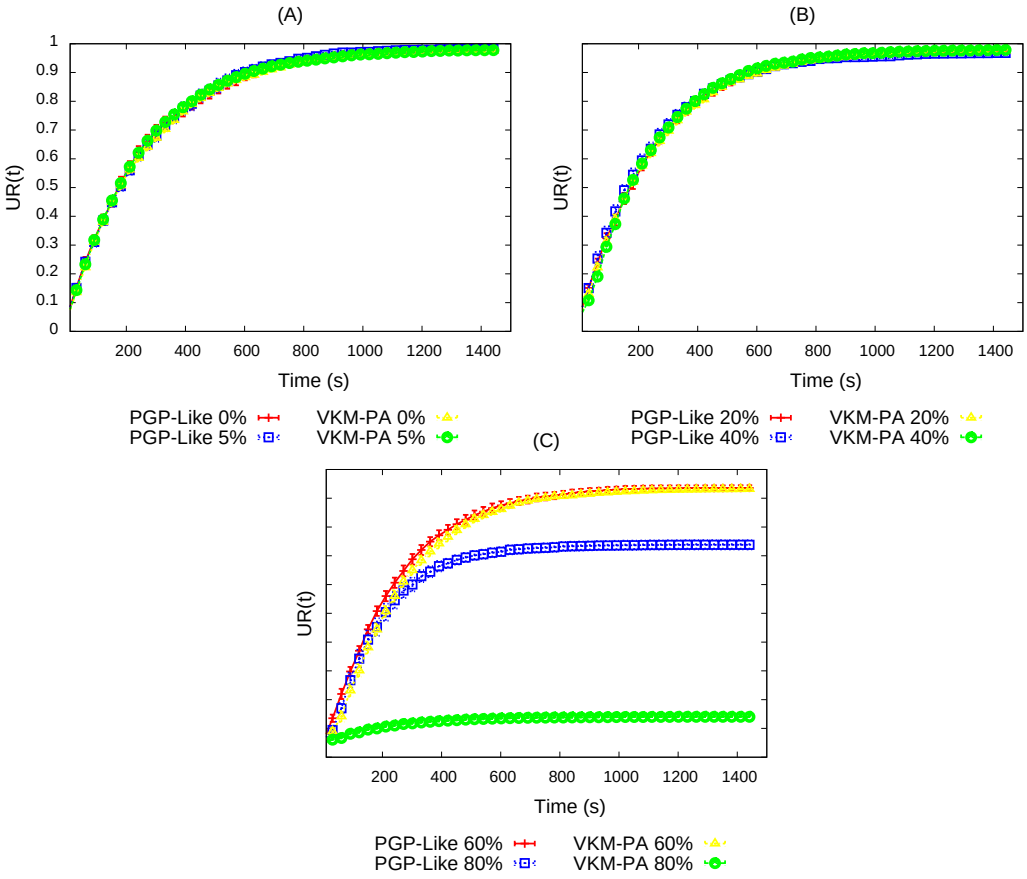


Figura 4.6: Alcançabilidade sob ataques de falta de cooperação.

virtual, i.e. como mostra a Figura 4.5 que apresenta simulações com 80% de nós egoístas e a conectividade aumentando de 5 até 20 certificados emitidos por nó (S). Note que, já com $|S = 10|$ o VKM-PA tem resultados melhores que o *PGP-Like*, com $|S = 15|$ e $|S = 20|$ a alcançabilidade é maior que 95%.

CAPÍTULO 5

PROTOCOLOS DE ROTEAMENTO

Os protocolos de roteamento para redes Ad Hoc podem ser divididos em protocolos pró-ativos, reativos e híbridos. Neste capítulo será exposto um resumo dos principais protocolos de roteamento existentes. Ainda nesse capítulo será introduzido o funcionamento dos protocolo de roteamento virtuais que serviram de base para a elaboração desse trabalho: *Virtual Routing Protocol* (VRP)[2] e *Virtual Distance Vector* (VDV)[40].

Os protocolos de roteamento pró-ativos mantém rotas atualizadas em suas tabelas de roteamento. Os nós realizam pedidos de rotas periodicamente de maneira a ter sempre uma rota disponível e atualizada para os demais nós da rede. A grande vantagem desse tipo de protocolo é o baixo atraso para o envio de dados, uma vez que a rota já estará disponível e atualizada no momento em que se desejar enviar os mesmos. Por outro lado, este tipo de protocolo acarreta uma grande sobrecarga à rede. O protocolo de roteamento pró-ativo mais conhecido é o DSDV [23].

O protocolo *Destination-Sequenced Distance-Vector* (DSDV) [36] é baseado em algoritmos de vetor de distância [6]. O mesmo utiliza o algoritmo de Bellman-Ford distribuído (DBF) [26]. Cada unidade de rede mantém uma tabela de roteamento que é periodicamente atualizada. A tabela de roteamento de cada unidade lista todos os possíveis destinos além do número de saltos até eles. Para manter as tabelas de roteamento constantemente atualizadas, cada unidade transmite periodicamente suas tabelas de roteamento quando há um número significante de novas atualizações disponível. A preferência na escolha da rota será por aquela com maior número de sequência, se houver mais de uma rota com o mesmo número de sequência, a preferência será por aquela com menor número de saltos.

Nos protocolos de roteamento reativos os nós realizam pedidos de rotas apenas sob demanda, isto é, apenas quando necessitam enviar dados para um determinado nó. A grande vantagem desse tipo de protocolo é a baixa sobrecarga da rede, uma vez que

são necessárias menos mensagens de controle pois não há a necessidade de manter as tabelas de roteamento constantemente atualizadas. Por outro lado, há um maior atraso no envio de dados, pois é necessário construir a rota da origem para o destino quando um nó desejar comunicar-se com outro nó. Os mais famosos e utilizados protocolos de roteamento reativos são o AODV e o DSR [4].

O *Ad Hoc On-Demand Distance Vector* (AODV) [37] é um protocolo puramente sob demanda. Os nós da rede não enviam suas tabelas de roteamento para outros nós, nem tentam descobrir ou manter rotas para outros nós. Os nós podem notificar sua existência aos seus vizinhos usando mensagens de *broadcast* local conhecidas como mensagens *Hello*. Para o descobrimento de rotas, o AODV confia aos nós intermediários da rede o armazenamento de informações em tabelas de roteamento dinamicamente estabelecidas. Quando um nó origem s deseja se comunicar com um nó destino d, o mesmo inicia um processo de descobrimento de rota para o nó d caso o mesmo não esteja na sua tabela de roteamento. O nó s inicia o processo de requisição de rota enviando um *Route Request* (RREQ) para seus vizinhos. Cada nó que retransmite o RREQ adiante, antes incrementa o contador de saltos e adiciona em sua tabela de roteamento o número de saltos para a origem e um salto para o nó que enviou o RREQ. Ao chegar no nó destino d, o RREQ será respondido com um *Route Reply* (RREP), de forma que este destino já possui uma rota para a origem. O RREP trafega pelo caminho reverso, que foi percorrido no envio do RREQ. Cada nó que transmite o RREP adiante, adiciona em sua tabela de roteamento o número de saltos para o destino e um salto para o nó que enviou o RREP.

No protocolo *Dynamic Source Routing* (DSR) [27], quando uma origem s deseja descobrir uma rota para um destino d, s deve criar uma mensagem RREQ que chegue até o destino d atrevés de inundação da rede. Esse pacote conterá em seu cabeçalho a sequência de saltos necessários para chegar até d. Quando um nó recebe uma mensagem de RREQ, ele anexa seu endereço na rota do cabeçalho da mensagem e a retransmite para seus vizinhos. Ao chegar no destino, d anexa seu endereço no final da rota do cabeçalho, inverte a rota contida no cabeçalho da mensagem e envia um RREP para a origem s com a rota invertida. A mensagem de RREP é enviada para s seguindo a rota contida no cabeçalho

da mensagem. Ao chegar na origem, s reverte a rota contida no cabeçalho da mensagem e passa a obter uma rota válida para d.

Protocolos de roteamento híbridos são protocolos que ora realizam pedidos de aquisição de rota de forma reativa, ora atualizam as rotas da tabela de roteamento de forma pró-ativa. Nesse tipo de protocolo, os nós levam em consideração a posição dos elementos da rede para decidir quando as rotas são atualizadas de forma pró-ativa ou de forma reativa.

O *Zone Routing Protoco*l (ZRP) [21] possui características de protocolos reativos e pró-ativos. O ZRP é baseado no conceito de zonas. Cada nó da rede possui sua zona de roteamento, que é limitada por σ saltos de distância. Os nós de uma zona podem ser do tipo periféricos ou interiores. Os periféricos são aqueles cuja distância para o nó central é de σ saltos. Nós interiores são aqueles cuja distância para o nó central é menor que σ. O roteamento para nós dentro da zona é pró-ativo enquanto o roteamento para nós fora da zona é reativo.

Virtual Routing Protocol (VRP)[2] e *Virtual Distance Vector* (VDV)[40] são dois protocolos híbridos. Ambos utilizam uma estrutura virtual que define a parte pró-ativa de aquisição e manutenção de rotas. Como esses protocolos serviram de base para esse trabalho ambos serão detalhados nas próximas seções.

5.1 Virtual Routing Protocol

O protocolo *Virtual Routing Protocol* (VRP)[2] foi criado com o objetivo de melhorar a taxa de entrega do protocolo DSR [27]. Da mesma forma que o DSR, o VRP usa *Source Routing*. *Source Routing* implica que cada mensagem de dados deve conter a rota que será percorrida na comunicação fim-a-fim.

O VRP usa uma *estrutura virtual* para definir os nós cujas rotas deverão ser requisitadas de maneira pró-ativa. Essa estrutura lógica sobre a rede é um grafo completamente independente das coordenadas físicas das unidades.

Seja o grafo $G = (V, A)$ uma estrutura virtual sobre a rede, onde V é o conjunto de unidades da rede e A é o conjunto de arestas que corresponde aos enlaces virtuais da rede. Os enlaces virtuais em A não são necessariamente bidirecionais. Para todo enlace

$(u, v) \in A$, u é chamada unidade espiã de v e v unidade espionada por u, i.e. $\forall(u, v) \in A, u$ é unidade espiã de v; $\forall(u, v) \in A, v$ é unidade espionada por u.

O conjunto das unidades que são espionadas por u é denominado S_u. O conjunto das unidades que são espiãs de u é denominado P_u. As unidades espiãs devem manter rotas atualizadas para suas unidades espionadas.

Todas as unidades conhecem a mesma estrutura virtual que deve ser construída e distribuída na inicialização da rede. Caso não seja possível fazê-lo na inicialização da rede, pode-se distribuir a estrutura virtual via *broadcast*, desde que seja garantido que as unidades da rede utilizem a mesma estrutura virtual.

A estrutura virtual mais apropriada deve ser selecionada pelo administrador da rede considerando propriedades como diâmetro, largura da bisseção e escalabilidade. Exemplos de estruturas virtuais são: o Anéis de Anéis - *Ring Of Rings (RoR)*, o Hipercubo, o Cubo de Ciclos de Conexos (CCC) e o 3D Torus.

5.1.1 Fase de Aquisição de Rota

Quando um nó s deseja se comunicar com um nó d, deve-se iniciar a fase de aquisição de rota. Seja N_s o conjunto de vizinhos de s. Se $d \in N_s$, a rota é trivial. Se $d \in P_s$ então s possui uma rota atualizada para d e esta fase também é trivial. Caso contrário, a aquisição de rota é feita em 2 passos: construção do caminho virtual e tradução para a rota real.

O caminho virtual é construído sobre a estrutura virtual G sem laços entre a origem s e o destino d. Nesta etapa não é necessária nenhuma comunicação uma vez que todas as unidades conhecem a mesma estrutura virtual. Os vértices que compõem o caminho virtual de s à d são chamados de saltos virtuais.

Após s construir o caminho virtual até d, s deve traduzir este caminho virtual em uma rota existente na rede. Para traduzir o caminho virtual, s envia uma mensagem que carrega o caminho virtual para d chamada de RTRANS. Como s já possuí um caminho real para o próximo salto virtual, s adiciona a rota que a mensagem RTRANS deve seguir no cabeçalho da mensagem. Quando o RTRANS chegar no próximo salto virtual, o mesmo

anexa a rota para o próximo salto virtual no cabeçalho da mensagem. Este processo se repete até que a mensagem chegue no destino d. No momento em que a mensagem RTRANS chegar ao destino, a rota contida no cabeçalho da mensagem é invertida e os laços contidos no caminho são removidos, de forma a enviar para a origem uma resposta sem laços. A mensagem RTRANS é então devolvida à origem, que ao recebê-la, armazena a rota para o destino e envia as mensagens de dados usando a rota estabelecida.

5.1.2 Fase de Manutenção de Rota

Devido à mobilidade, enlaces podem se quebrar, alterando as rotas armazenadas nas tabelas de roteamento dos nós. Com isso, unidades podem perder a sua conexão com um nó vizinho. Uma rota $R = \{r_0, r_1, ..., r_m\}$ se torna desconexa quando duas unidades consecutivas perdem sua conexão, seja pela mobilidade, aparição de um obstáculo entre a comunicação ou uma falha isolada.

A mensagem de quebra de enlace *Route Error* (RERR) pode ser enviada tanto na fase de **tradução de rota** quanto na fase de **envio de dados**. Na fase de **tradução de rota**, quando uma unidade r_i tenta enviar uma mensagem de RTRANS para outra unidade r_{i+1} e r_i descobre uma quebra de enlace, r_i envia uma mensagem RERR para a origem e também para a última espiã pertencente ao caminho virtual. Dessa forma, a última unidade espiã poderá iniciar o processo de atualização de rota para a unidade espionada.

Quando uma unidade r_i tenta **enviar dados** para outra unidade r_{i+1} e r_i descobre uma quebra de enlace, r_i envia uma mensagem *Route Error* (RERR) para a origem. Neste caso não é possível informar a última unidade espiã, pois a mensagem de dados não contém o caminho virtual original em seu cabeçalho.

Quando a origem recebe uma mensagem RERR, a mesma deve construir uma nova rota para o destino se ainda for desejado. A nova rota pode ser construída utilizando outro caminho virtual ou através de inundação na rede com uma mensagem de RREQ semelhante ao protocolo DSR [27].

5.1.3 Fase de Atualização de Rota

A fase de atualização de rota é feita pelas unidades espiãs. Quando uma unidade espiã u recebe uma mensagem de RERR informando que a rota utilizada para outra unidade não é mais válida, a mesma procura pró-ativamente por rotas para todas as unidades $\in P_u$. A unidade espiã u envia uma mensagem de RREQ especial destinada a α ou outro símbolo que se diferencie dos identificadores das demais unidades da rede. α também pode ser o endereço de *broadcast* da rede.

Quando uma unidade v recebe um RREQ destinado a α, ela verifica se o endereço de origem $\in S_v$. Caso $u \in S_v$, u responde ao RREQ. Caso seja desejável, a fase de atualização de rotas pode ser periódica, de maneira que as unidades da rede atualizem as rotas para suas unidades espionadas em um intervalo δ. Quando uma unidade recebe um RERR, ela atualiza o seu intervalo δ para $tempoAtual + \delta$. Assim, evita-se que uma nova mensagem de descoberta de rota seja enviada em um intervalo menor do que δ.

5.2 Virtual Distance Vector

Ao contrário do VRP, o VDV não utiliza *Source Routing*. Ao invés disso, as rotas são configuradas pelos nós da rede pelo uso de *Distance Vectors*. Isso implica que nós intermediários devem manter registros de quantos saltos deverão ser realizados por uma mensagem para ser entregue em um determinado destino d. Os nós também necessitam manter a informação de qual é o próximo nó por onde a mensagem deverá ser enviada para eventualmente ser recebida por d.

O protocolo foi criado com o objetivo de acelerar o tempo para o envio de mensagens do protocolo AODV, evitando o atraso que geralmente ocorre em protocolos reativos devido à fase de descoberta de rotas. Dessa forma, a grande diferença para o VRP é a ausência de uma fase de tradução de rota como a mensagem RTRANS.

5.2.1 Roteamento de Mensagem de Dados

Cada nó da rede confia em seus vetores de distância adquiridos na fase pró-ativa do protocolo. Dessa forma, seja g o grau da estrutura virtual usada pelo VDV, cada nó deve manter pelo menos g vetores de distância, onde cada vetor de distância é um par $\{destino, PróximoNó(destino)\}$. Portanto, o destino dos g vetores de distância que cada nó mantém armazenado nada mais é do que as unidades espionadas por esse nó. Outros destinos contidos nos vetores de distância consistem em próximos nós para atingir um determinado destino. O protocolo ainda utiliza uma Tabela de Pacotes de Dados Recentes (TPDR). Cada elemento dessa tabela é uma tripla $\{PacketId, Origem, Destino\}$ e é mantida na tabela até expirar.

A funcionalidade do roteamento se segue da seguinte forma: quando um pacote de dados é criado na origem s para ser enviado para o destino d, a mesma calcula um caminho na estrutura virtual e envia a mensagem para uma de suas unidades espionadas. As unidades espionadas, que recebem uma mensagem de dados, computam um caminho na estrutura virtual para d e enviam a mensagem para a sua unidade espionada mais próxima do destino. Isso ocorre sucessivamente até a mensagem ser recebida por d.

Durante o roteamento de uma mensagem de dados pela rede, pode haver a formação de laços. Para evitar isso, cada vez que uma mensagem de dados for roteada por um determinado nó, ele deve criar uma entrada na tabela TPDR. Caso a mensagem de dados passe novamente por esse nó, o mesmo deverá atualizar o seu vetor de distância para d com as novas informações de roteamento. Dessa forma os próximos pacotes que passarem por esse nó seguirão automaticamente por esse caminho, e dessa forma, os laços formados durante o roteamento das mensagens serão removidos.

Enquanto o nó que está roteando a mensagem encontrar um vetor de distância válido, a mensagem viaja pela rede. Dessa forma, um pacote roteado pela rede, elimina os laços e otimiza as tabelas de roteamento dos nós intermediários para possíveis pacotes que virão em uma rajada subseqüente a esse pacote.

5.2.2 Manutenção de Rotas

Caso uma mensagem que está sendo roteada pela rede não tenha um vetor de distância válido para continuar seu roteamento, o nó que detectou a quebra de enlace pode realizar um *detour* (desvio do roteamento para um outro nó) [40], enviando a mensagem para outra unidade espionada. Caso esse nó não possa realizar mais um *detour*, o mesmo inicia uma inundação na rede na tentativa de descobrir uma rota para d. Caso uma rota não seja encontrada, a mensagem é descartada. Finalmente, além da tentativa de encontrar uma rota para d, o nó que ficou impossibilitado de executar o *detour* deve enviar uma mensagem de erro para a unidade espiã, de maneira que a mesma atualize suas rotas evitando uma nova inundação na rede ou um novo descarte.

CAPÍTULO 6

RESULTADOS DO VKM EMBUTIDO NO ROTEAMENTO

Nesse capítulo, será apresentado um estudo sobre o impacto no desempenho dos protocolos de roteamento *Virtual Routing Protocol* [2] e *Virtual Distance Vector* [40] quando os mesmos incorporam o sistema de gerenciamento de chaves virtual VKM embutido diretamente no roteamento.

O simulador *Network Simulator* 2 (NS-2) [35], em sua versão 33, foi usado para avaliar o impacto na performance do roteamento dos protocolos virtuais sob o uso do VKM. A versão 33 foi escolhida por ser, no momento da implementação dos protocolos de roteamento virtuais, a versão mais recente e recomendada em [35]. Os resultados são médias de trinta e cinco simulações com intervalo de confiança de 95%. A estrutura virtual escolhida foi o RoR com três anéis. Cada nó mantém rota pró-ativamente para 5 outros nós.

Tabela 6.1: Cenários das simulações do VKM implementado nos protocolos de roteamento virtuais (VRP e VDV)

Parâmetros	**Valor**
Dimensão da rede	1000 x 1000 metros e 1500 x 300 metros
Alcance da transmissão	250 metros
Nós	51, 75 e 108
Modelo de mobilidade	*random waypoint*
Velocidade máxima	0, 2, 4, 6, 8, 10, 12, 14, 16, 18 e 20 m/s
Tempo de pausa máximo	0 e 10 segundos
Mensagens enviadas por segundo	4 mensagens de dados enviadas por segundo
Número de conexões CBR	20 conexões CBRs estabelecidas ao mesmo tempo
Tempo de Simulação	600 segundos
Modelo de Propagação	*two-ray ground reflection*
Protocolo de Acesso ao Meio	IEEE 802.11

6.1 Impacto do VKM sobre o Roteamento

Considerando a existência de uma estrutura virtual para estabelecer as regras de confiança na rede, o *Virtual Routing Protocol* (VRP) e o *Virtual Distance Vector* (VDV) também

utilizam uma estrutura virtual que define a parte pró-ativa de aquisição e manutenção de rotas. Um dos objetivos desse trabalho é avaliar o impacto que a utilização do VKM causaria quando implementado em conjunto com os protocolos de roteamento virtuais. Não cabe a este trabalho analisar se os valores sao aceitáveis ou não, mas sim relatar os valores encontrados. Cabe os usuário escolher se vale a pena usar o esquema de gerenciamento de chaves para fornecer segurança ao protocolo denegrindo a eficiência dos algoritmos, ou manter a eficiência e não usar a criptografia.

Nas próximas seções será analisado o impacto no desempenho dos protocolos de roteamento VRP [2] e VDV [40] quando os mesmos incorporam o sistema de gerenciamento de chaves virtual VKM implementado diretamente no roteamento. Para realizar esse estudo, foi implementado no simulador *Network Simulator* 2 (NS-2) [35], em sua versão 33, os protocolos de roteamento VRP e VDV. A implementação dos protocolos foi realizada seguindo as especificações descritas em [2] e em [40] respectivamente. A validação da implementação do VRP no NS-2 pode ser encontrada no Apêndice A e a validação da implementação do VDV no NS-2 pode ser encontrada no Apêndice B. Os parâmetros usados nas simulações estão na Tabela 6.1. Todos os protocolos foram simulados considerando tempo de pausa máximo de 0 e 10 segundos e serão referenciados nos gráficos como $P0$ e $P10$ respectivamente. Os cenários diferem no tamanho de rede: 1000x1000 (rede quadrada) para 1500x300 (rede retangular); e no número de unidades em cada cenário: 51, 75 e 108. É observado o impacto na taxa de entrega (% de pacotes dados entregues), na sobrecarga gerada (x pacotes roteados para cada pacote de dado efetivamente entregue) e o atraso no envio de dados (medido em milisegundos).

6.1.1 Análise do Impacto no Roteamento do VRP

Para analisar o impacto ao embutir o esquema de gerenciamento de chaves VKM diretamente no protocolo de roteamento VRP, foi removida a descoberta de rota através de *flooding* realizada de uma origem s para um destino d ao tentar enviar dados para esse destino. O impacto da retirada das mensagens de RREQs na rede (semelhante a descoberta de rota do DSR) encontra-se no Apêndice C. O motivo da remoção das men-

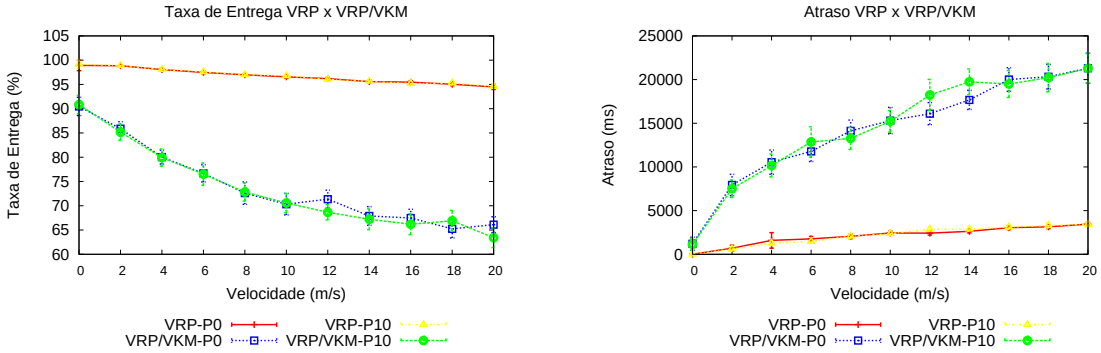
sagem de RREQs é o interesse em forçar o envio de dados utilizando as rotas definidas pela estrutura virtual. Assim, mesmo quando não é possível traduzir uma rota virtual em uma rota real na rede, as unidades deverão atualizar suas rotas e reenviar os pedidos de tradução de rotas, e dessa forma, mantendo as mensagens de dados trafegando pelas rotas definidas pela estrutura virtual.

Em uma comunicação fim-a-fim, uma origem s precisa solicitar os certificados das demais unidades virtuais até o destino d, de forma a obter e validar o certificado de d transitivamente. Assim, d poderia cifrar a mensagem de RTRANS antes de devolver a mensagem para a origem s. Ao receber a mensagem RREQ de d, s terá a certeza de que d é realmente quem diz ser, e pode então criptografar as mensagens de dados para serem recebidas apenas por d.

Se o caminho da estrutura virtual for composto por n nós, a origem s deve requisitar $n - 2$ certificados, pois a origem não precisa requisitar o seu próprio certificado nem o do próximo nó pertencente ao caminho virtual. Esse procedimento deve ser feito apenas no primeiro processo de tradução de rota. Uma nova requisição de certificados será feita quando os certificados forem revogados ou quando a rota deixar de ser válida, seja por quebra de enlace ou por envelhecimento da mesma. Nas simulações realizadas no NS-2, o tempo de vida de uma rota é o mesmo tempo de revogação dos certificados. Assim é possível avaliar o impacto que a requisição de certificados causa no roteamento do VRP.

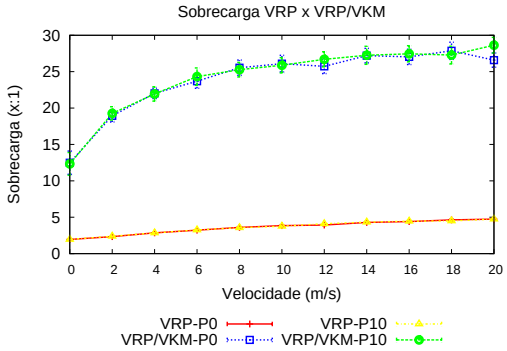
A Figura 6.1 mostra que ao requisitar os certificados das unidades que formam o caminho virtual, a taxa de entrega cai. A taxa de entrega que antes era sempre superior a aproximadamente 94%, se encontra abaixo de 60% quando as unidades se movimentam à 20m/s. Isso ocorre devido a uma sobrecarga 6 vezes maior gerada na rede, causada pela requisição dos certificados.

Da mesma forma, pelo fato de uma origem necessitar obter todos os certificados de todas as unidades que formam o caminho virtual antes de enviar dados, o atraso no envio de mensagens chega a ser 10 vezes maior em comparação com a especificação original do VRP. Ainda é possível ocorrer o caso de apenas um certificado não ser recebido para uma comunicação fim-a-fim, e o mesmo deverá ser requisitado novamente. Nesse caso, mais



(a) Taxa de Entrega

(b) Atraso

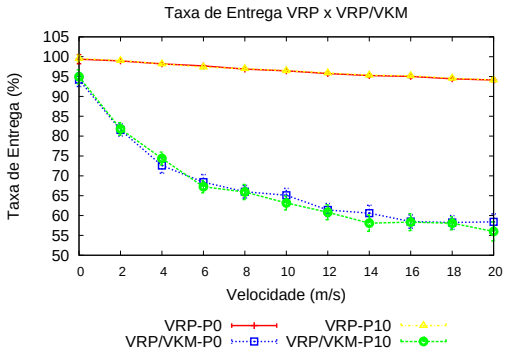


(c) Sobrecarga

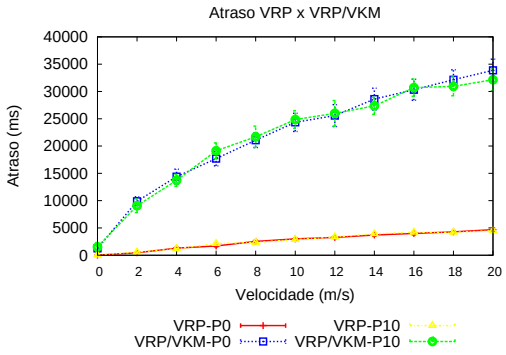
Figura 6.1: VRP X VRP/VKM - Cenário 1000mx1000m com 51 nós

retransmissões ocorrerão na rede, aumentando a sobrecarga. A sobrecarga gerada tem impacto direto na taxa de entrega do protocolo e no tempo necessário para o envio de dados.

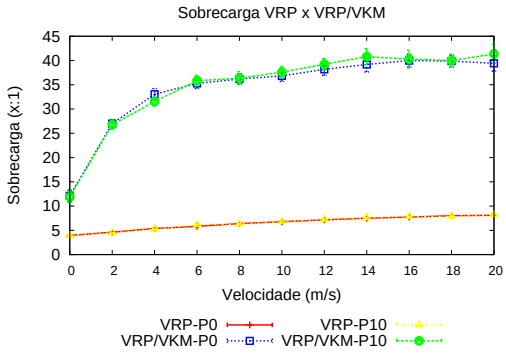
A Figura 6.2 mostra uma queda acentuada na taxa de entrega, ficando abaixo de 60% quando os nós atingem a velocidade de 20m/s. O atraso no envio de mensagens chega a ser 6 vezes maior, com um salto de aproximadamente 5000ms para 35000ms. Por fim a sobrecarga apresenta novamente um aumento de quase 6 vezes em relação a implementação original do VRP. A medida que o número de unidades da rede aumenta, a mesma se torna mais congestionada. Isso ocorre por que haverá mais unidades realizando requisições de atualização de rota.



(a) Taxa de Entrega

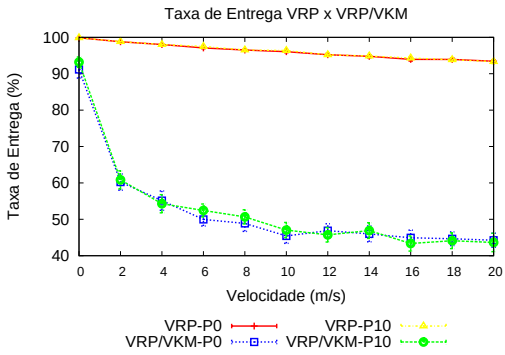


(b) Atraso

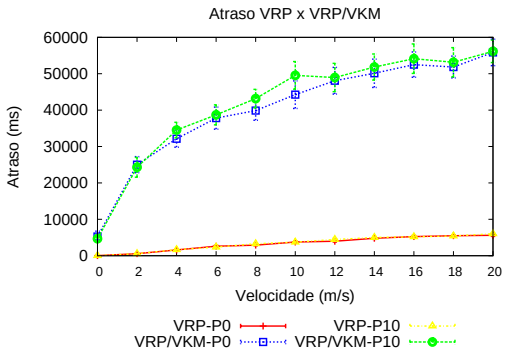


(c) Sobrecarga

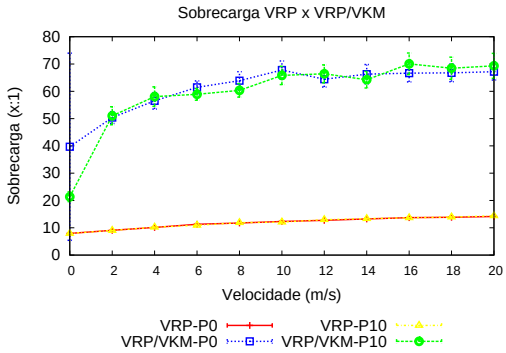
Figura 6.2: VRP X VRP/VKM - Cenário 1000mx1000m com 75 nós



(a) Taxa de Entrega



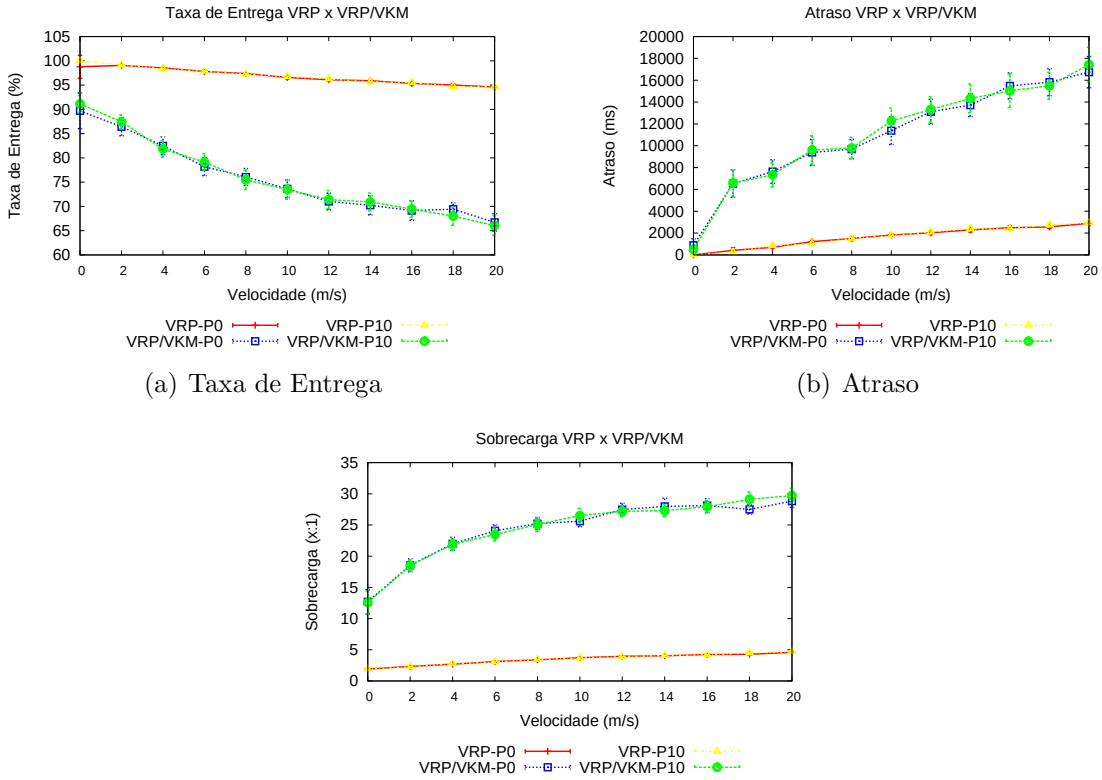
(b) Atraso



(c) Sobrecarga

Figura 6.3: VRP X VRP/VKM - Cenário 1000mx1000m com 108 nós

Com a rede altamente congestionada como mostrado na Figura 6.3, o uso do VKM no protocolo VRP apresenta uma grande queda na taxa de entrega. A mesma cai para pouco mais de 40% com os nós atingindo uma velocidade de 20m/s. A sobrecarga gerada chega a ser 7 vezes maior, causando novamente um aumento de 6 vezes no atraso de envio de mensagens, podendo chegar a até 60000ms.



(a) Taxa de Entrega

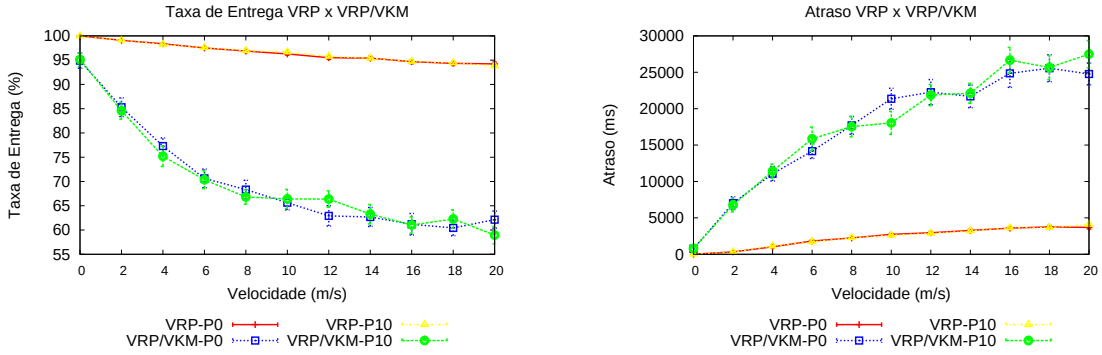
(b) Atraso

(c) Sobrecarga

Figura 6.4: VRP X VRP/VKM - Cenário 1500mx300m com 51 nós

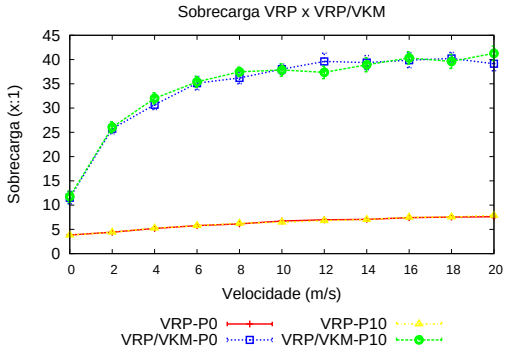
Como visto na Figura 6.4, os resultados obtidos com 51 nós em uma rede retangular se assemelham aos resultados obtidos com um espalhamento uniforme da rede. A taxa de entrega do VRP utilizando o VKM mantém-se acima de 65% quando os nós atingem a velocidade máxima de 20m/s. A sobrecarga apresenta um acréscimo de até 6 vezes em relação ao VRP original e o atraso no envio de mensagens mostra um aumento de até 8 vezes, subindo de aproximadamente 2500ms para até 18000ms.

Na Figura 6.5 podemos observar que com 75 nós em uma rede retangular os resultados são similares aos obtidos com um espalhamento uniforme dos mesmos. A taxa de entrega cai de aproximadamente 95% para menos de 65% quando os nós se movimentam à 20m/s.



(a) Taxa de Entrega

(b) Atraso



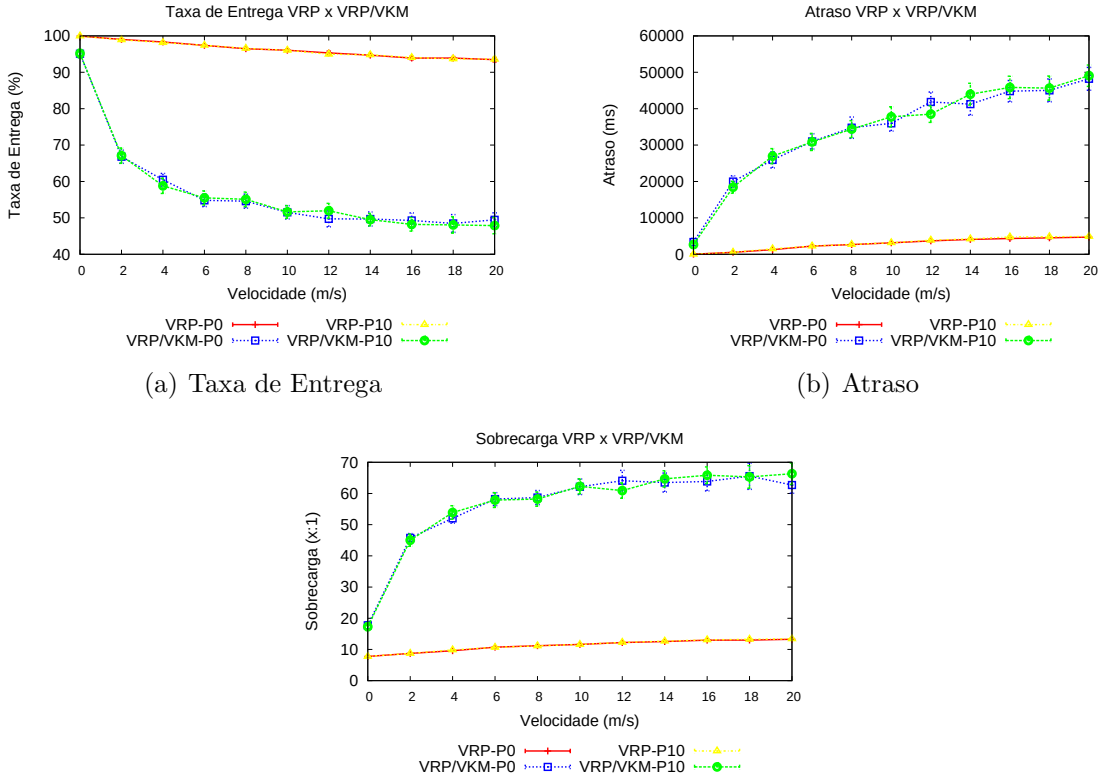
(c) Sobrecarga

Figura 6.5: VRP X VRP/VKM - Cenário 1500mx300m com 75 nós

O atraso no envio de mensagens novamente aumenta de aproximadamente 4000ms para mais de 25000ms.

Por fim, a Figura 6.6 mostra que com 108 nós em uma rede retangular, a taxa de entrega do VRP com o VKM embutido cai para aproximadamente 50%, enquanto a taxa do VRP original mantém-se acima de 90%. O atraso no envio de mensagens tem um aumento de até 10 vezes, novamente justificada devido à sobrecarga causada na rede, que chega a ser quase 7 vezes maior do que a causada pelo VRP original.

Os resultados mostram que a requisição de certificados pela origem em uma comunicação fim-a-fim causam uma grande sobrecarga na rede. Essa mesma sobrecarga na rede causada pela requisição dos certificados mostra-se consideravelmente maior do que aquela apresentada na concepção original do VRP, e dessa forma, afeta o atraso no envio de dados e também a taxa de entrega de mensagens. Portanto, o impacto de usar o VKM em cima do VRP é aumentar o atraso no envio de mensagens, aumentar a sobrecarga na rede e diminuir a taxa de entrega de mensagens, independente do cenário utilizado.



(a) Taxa de Entrega

(b) Atraso

(c) Sobrecarga

Figura 6.6: VRP X VRP/VKM - Cenário 1500mx300m com 108 nós

6.1.2 Análise do Impacto no Roteamento do VDV

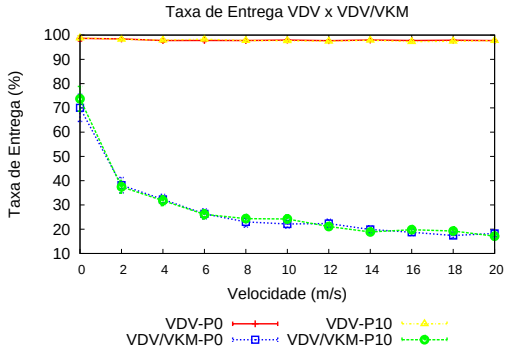
Para analisar o impacto ao embutir o esquema de gerenciamento de chaves VKM diretamente no protocolo de roteamento VDV, a exemplo do que foi feito no VRP, é necessário remover a descoberta de rota através de *flooding*. O impacto na retirada das mensagens de RREQs na rede encontra-se no Apêndice D. O motivo da remoção da mensagem de RREQs é o interesse em forçar o envio de dados utilizando as rotas definidas pela estrutura virtual.

Diferente do VRP onde o envio de dados é dividido em duas etapas (tradução do caminho virtual e envio usando a rota real), o envio de dados do VDV é feito imediatamente, deixando a cargo das unidades intermediárias qualquer necessidade de reconstrução de rotas. Entretanto, como o RREQ realizado por unidades intermediárias após o esgotamento de possíveis desvios é removido da rede, sempre que uma mensagem não possa ser delegada a um próximo nó ela será descartada. Portanto, espera-se uma queda na taxa de entrega apenas removendo o uso de RREQs na rede.

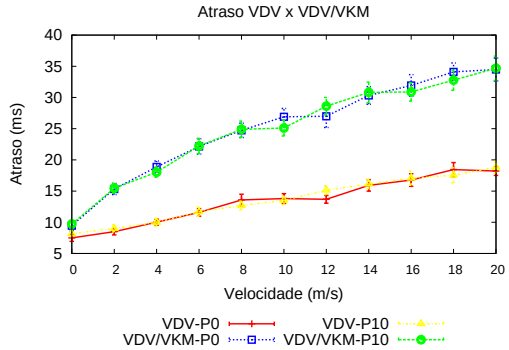
Cada nó virtual confia na sua unidade espionada, e assim pode criptografar os dados da mensagem de forma que apenas aquela unidade consiga descobrir o conteúdo da mesma. A origem envia os dados criptografados para a próxima unidade virtual, e transitivamente, o destino recebe os dados de uma unidade que confia nele.

Não é necessária nenhuma requisição de certificados ao embutir o VKM no VDV, uma vez que o envio de dados é feito imediatamente para os próximo salto. Nas simulações realizadas no NS-2, foi removido o uso de mensagens *Hello*, pois é desejável que as mensagens de dados trafeguem apenas pelas rotas definidas pela estrutura virtual.

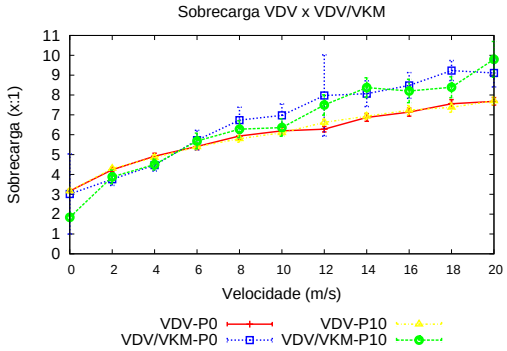
A Figura 6.7 mostra que com a remoção de descobertas de rotas através de *floodings* e com a remoção de mensagens de *Hello*, a taxa de entrega cai para menos de 20% quando as unidades atingem a velocidade de 20m/s. Ainda, o atraso no envio de mensagens praticamente dobrou, indo de pouco menos de 20ms para aproximadamente 35ms. A sobrecarga se mantém similar devido ao aumento de mensagens de erro e de atualização de rotas.



(a) Taxa de Entrega



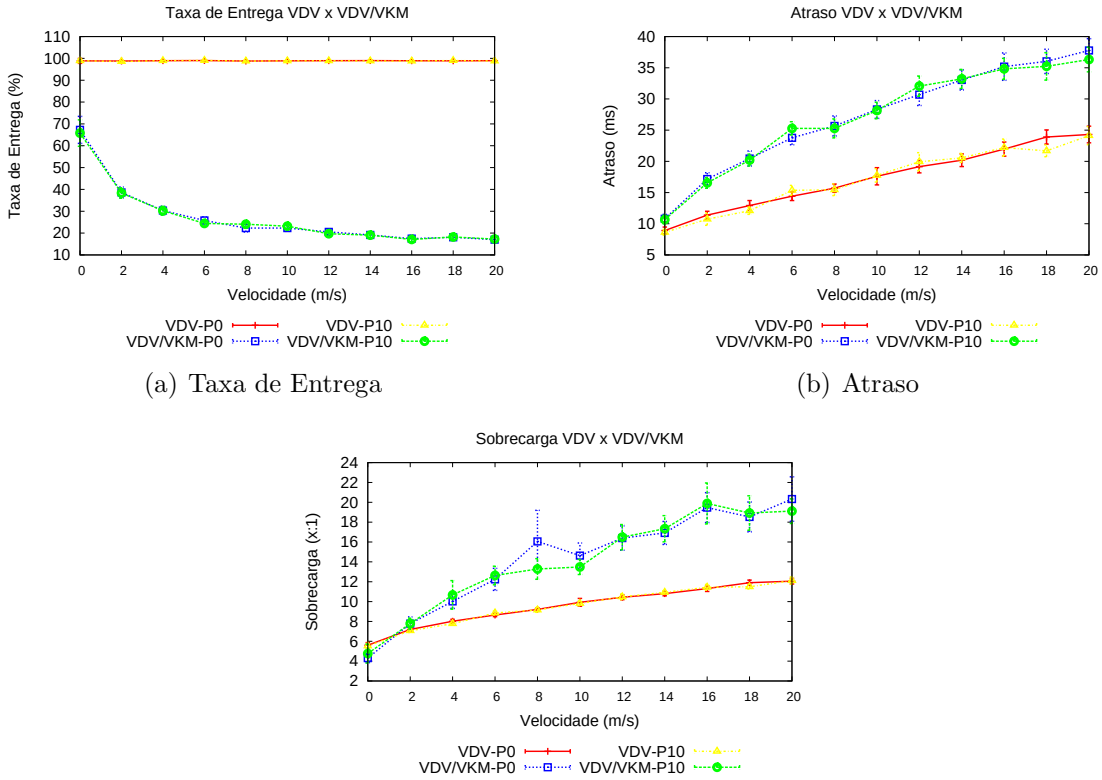
(b) Atraso



(c) Sobrecarga

Figura 6.7: VDV X VDV/VKM - Cenário 1000mx1000m com 51 nós

A Figura 6.8 mostra que a exemplo do cenário com 51 nós, com 75 nós a taxa de entrega cai para menos de 20% quando as unidades atingem a velocidade de 20m/s. Da mesma forma, o atraso no envio de mensagens praticamente dobrou, indo de pouco mais de 20ms para aproximadamente 40ms. Mesmo com a remoção de mensagens *Hello*, a sobrecarga aumentou devido ao aumento de mensagens de erro e requisição de atualização de rotas.



(a) Taxa de Entrega

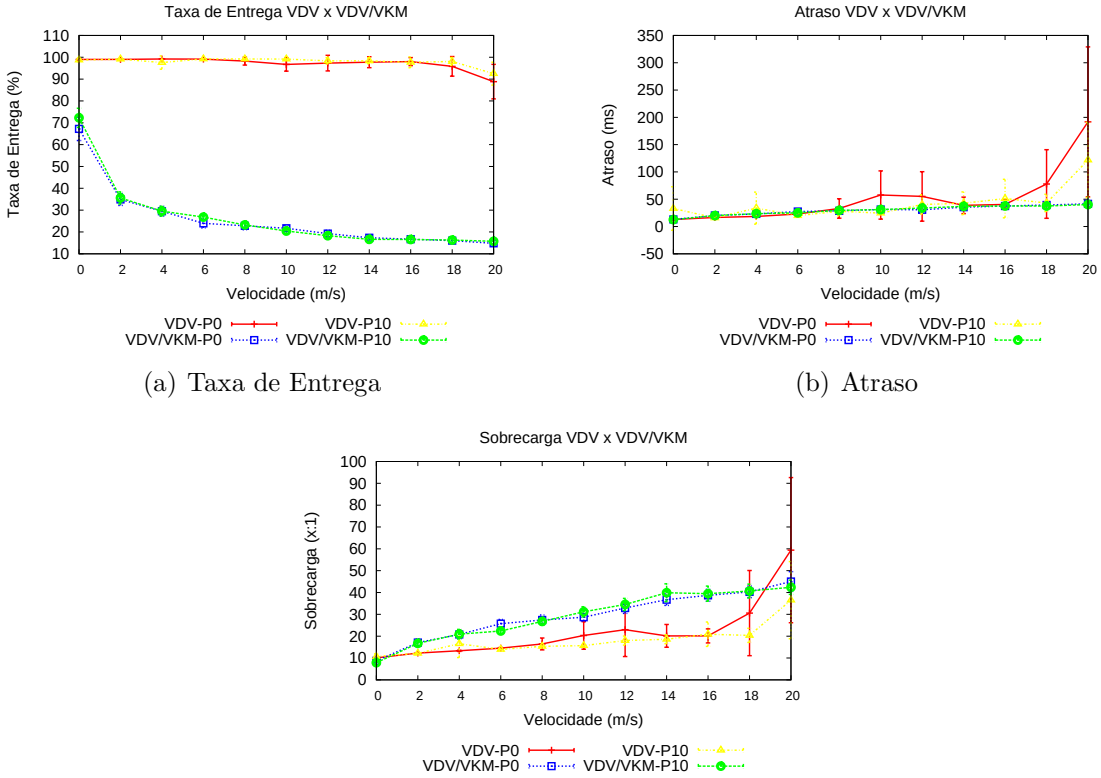
(b) Atraso

(c) Sobrecarga

Figura 6.8: VDV X VDV/VKM - Cenário 1000mx1000m com 75 nós

Em um cenário altamente congestionado com 108 nós, de acordo com a Figura 6.9 a taxa de entrega na implementação com o VKM cai para menos de 20% quando as unidades atingem a velocidade de 20m/s. Porém, devido ao congestionamento da rede e da implementação com o VKM utilizar apenas rotas estabelecidas pela estrutura virtual, o atraso no envio de mensagens fica praticamente constante em menos de 50ms, mesmo com o aumento da velocidade dos nós. Já no VDV original, com o aumento do número de unidades na rede, o atraso no envio de mensagens dobra quando as unidades passam da velocidade de 18m/s, chegando a ser próximo a 150ms quando os nós atingem 20m/s. Mesmo com a remoção de mensagens *Hello*, a sobrecarga aumentou devido ao aumento

de mensagens de erro e requisição de atualização de rotas.



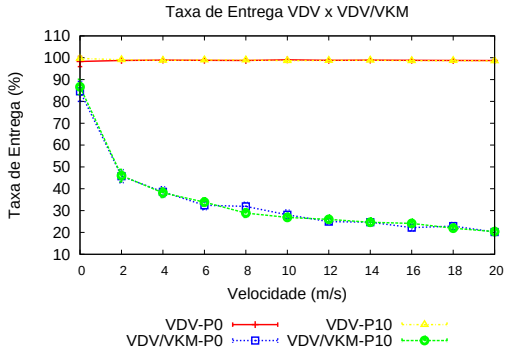
(a) Taxa de Entrega

(b) Atraso

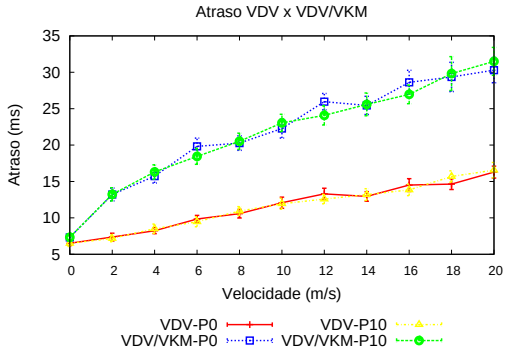
(c) Sobrecarga

Figura 6.9: VDV X VDV/VKM - Cenário 1000mx1000m com 108 nós

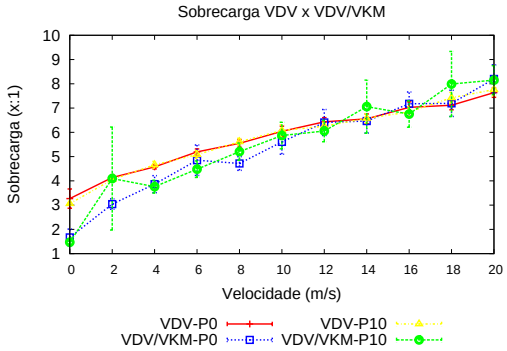
Quando comparado a implementação do VKM em um cenário com 51 nós em uma rede retangular, de acordo com a Figura 6.10 o comportamento da taxa de entrega e da sobrecarga foi semelhante ao cenário com 51 nós espalhados em uma rede quadrada. O atraso no envio de mensagens praticamente dobrou: de aproximadamente 15ms para aproximadamente 30ms. Por outro lado, a sobrecarga causada na rede se manteve similar à implementação original do VDV.



(a) Taxa de Entrega



(b) Atraso



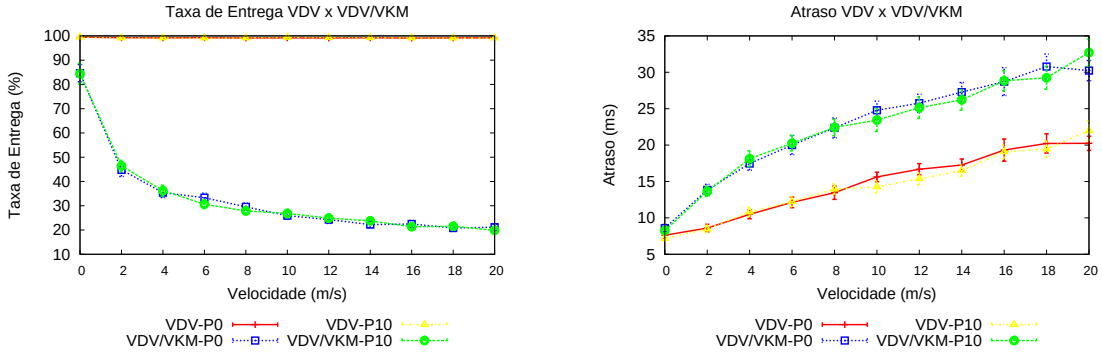
(c) Sobrecarga

Figura 6.10: VDV X VDV/VKM - Cenário 1500mx300m com 51 nós

Já para a implementação do VKM em um cenário com 75 nós em uma rede retangular, de acordo com a Figura 6.11 o comportamento da taxa de entrega e da sobrecarga foi semelhante ao cenário com 75 nós espalhados de maneira uniforme. O atraso no envio de mensagens subiu de aproximadamente 20ms para aproximadamente 35ms. Ainda, a sobrecarga causada na rede se manteve maior em relação à implementação original do VDV.

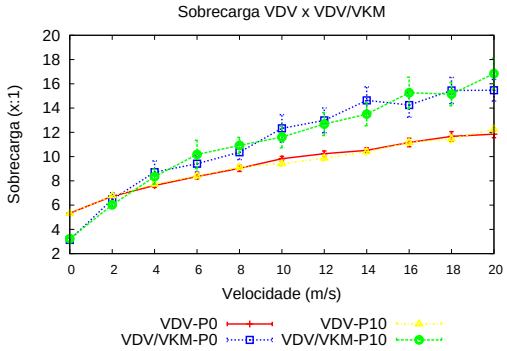
Finalmente, de acordo com a Figura 6.12 a implementação com o VKM apresenta uma taxa de entrega menor do que 20% quando as unidades se movimentam com uma velocidade de 20m/s. A sobrecarga na rede cresceu mais do que o dobro devido ao número de mensagens de erro e de atualização de rotas, o que também ocasionou um aumento no atraso de envio de mensagens.

Os resultados apresentados nessa Sessão mostram o que acontece com o VDV quando não é realizada nenhuma descoberta de rotas pelos nós intermediários, além de ser removido o uso de mensagens *Hello*. Ao se remover o envio de mensagens de RREQ re-



(a) Taxa de Entrega

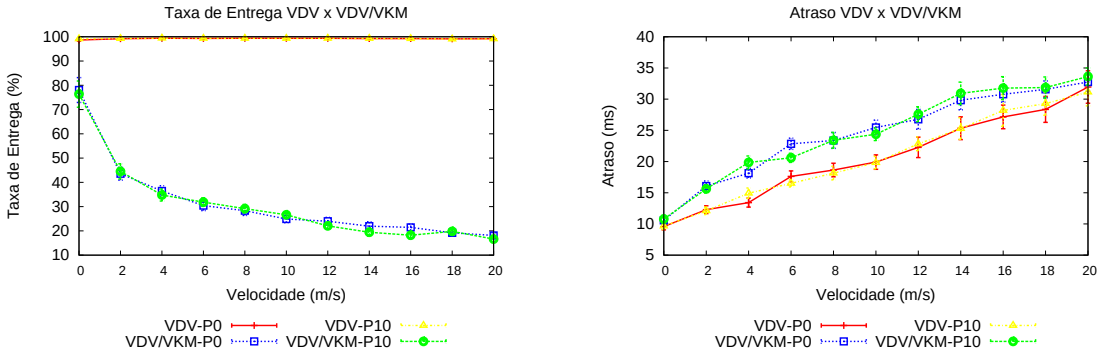
(b) Atraso



(c) Sobrecarga

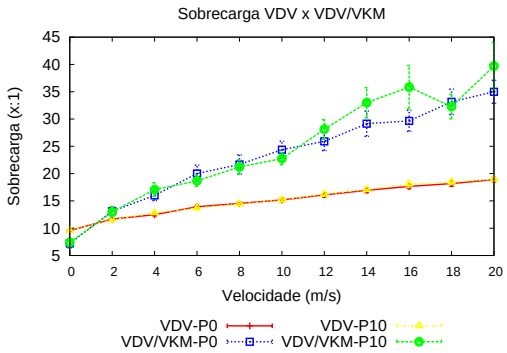
Figura 6.11: VDV X VDV/VKM - Cenário 1500mx300m com 75 nós

alizado pelos nós intermediários quando os mesmos não possuem mais nenhum *detour*, uma grande queda na taxa de entrega foi observada. Posteriormente, ao confiar apenas na parte pró-ativa do protocolo para estabelecer o envio de dados, removendo também as mensagens *Hello*, a taxa de entrega cai drasticamente. Dessa forma, a remoção de RREQs combinada com a remoção de mensagens *Hello* afeta consideravelmente a taxa de entrega de mensagens do VDV. Ao se utilizar o VKM embutido diretamente no VDV, é compreensível que para este protocolo específico, a taxa de entrega cai drasticamente, pois para perfeito funcionamento do VDV é necessário haver o envio de RREQs pelos nós intermediários de cada comunicação fim-a-fim.



(a) Taxa de Entrega

(b) Atraso



(c) Sobrecarga

Figura 6.12: VDV X VDV/VKM - Cenário 1500mx300m com 108 nós

CAPÍTULO 7

CONCLUSÕES

As redes MANETs foram criadas idealizando aplicações distribuídas de forma auto-organizável em ambientes onde o meio de comunicação é sem-fio e o roteamento de mensagens nessas redes é realizado de forma cooperativa e distribuída entre os próprios nós da rede. Essas características fazem com que as redes Ad Hoc apresentem não só todos os problemas de segurança existentes em redes cabeadas e em redes sem-fio estruturadas, como também novos desafios.

Diversos tipos de ataques podem ser realizados contra as MANETs em diversas camadas de rede. Para prover segurança contra ataques nessas redes é necessário o uso de criptografia. A criptografia é considerada a principal técnica para garantir segurança em redes e pode ser tanto do tipo simétrica ou do tipo assimétrica. Na criptografia simétrica, os nós utilizam a mesma chave para realizar a cifração de dados. Na criptografia assimétrica, os nós possuem duas chaves diferentes, sendo uma para cifrar mensagens e a outra para decifrar. A tarefa de administrar essas chaves é definida por um esquema de gerenciamento de chaves, que define o armazenamento, a distribuição, a proteção e a revogação das mesmas.

Para ser utilizado em MANETs, um esquema de gerenciamento de chaves públicas deve ser distribuído e auto-organizado. É possível classificar os esquemas de gerenciamento de chaves para MANETs em: baseado em identidade, baseado em cadeias de certificados, baseado em clusters, baseado em pré-distribuição e baseado em mobilidade. Dentre todos os tipos de esquemas de gerenciamento de chaves públicas, os que melhores se adequam para aplicações nessas redes parecem ser os baseados em certificados. O *PGP-Like*, é o principal esquema de gerenciamento de chaves públicas para MANETs baseado em cadeias de certificados. Entretanto, o *PGP-Like* é altamente vulnerável a ataques de personificação. A funcionalidade do *PGP-Like* é comprometida, mesmo com apenas 5%

de nós personificados ou identidades falsas na rede.

O esquema de gerenciamento de chaves baseado em grupos - *Group-based Key Management* (GKM) também é baseado em certificados e seus usuários formam pequenos grupos, nos quais todos os nós tem o mesmo papel sem a necessidade de haver um líder. Estes grupos são formados com base no relacionamento dos usuários que formam uma rede de grupos. Essa rede é usada para realizar todas as operações de gerenciamento de chaves e os nós precisam ser um membro de um grupo para fazer parte do sistema. O GKM consegue oferecer até 90% de segurança contra ataques do tipo *Sybil*, sem haver distinção entre o tipo de ataque *Sybil*. Sua performance é a mesma independente se os atacantes criam identidades falsas ou se os mesmos personificam outras unidades existentes na rede.

Este trabalho introduziu um novo sistema de gerenciamento de chaves públicas para MANETs, o *Virtual Key Management System* (VKM), que é 100% resistente a ataques de criação de identidades falsas. O VKM faz uso de uma estrutura virtual para indicar a confiança entre os nós e a formação de cadeias de certificados, sendo o primeiro esquema de gerenciamento de chaves completamente baseado em virtualização.

O VKM é um esquema de gerenciamento de chaves muito flexível, podendo ser configurado de duas maneiras diferentes: VKM-RA e VKM-PA. Usando o VKM-RA, os nós seguem as regras da estrutura virtual para emitir certificados e autenticar chaves públicas. Como mostrado nos resultados obtidos, nós atacantes que tentem personificar unidades de uma maneira desorganizada não conseguem comprometer o funcionamento da rede. Isso, considerando que esses nós atacantes consigam descobrir as regras que estabelecem a estrutura virtual, a qual é uma informação privilegiada para alguns nós da rede. O VKM-RA é capaz de completar corretamente 80% de todas as requisições de autenticação com 5% de nós personificados na rede. Mesmo com 20% de nós personificados na rede, o VKM-RA consegue autenticar em torno de 50% das cadeias de certificados.

Quando comparado o VKM com o GKM é possível concluir que o segundo é menos afetado por ataques de personificação. No GKM, a efetividade só é afetada com 40% atacantes. Neste caso, a métrica NCA é aproximadamente 40% com o tamanho de grupo $G = 3$, enquanto que a mesma se mantém com 90% com $G = 6$. Já no VKM, em um

cenário com 40% de nós atacante e número de certificados emitidos $S = 6$, o valor de NCA é aproximadamente 13%, enquanto que quando a quantidade de atacantes é 5%, NCA é aproximadamente 78%. Essa diferença ocorre devido a necessidade de duas cadeias de certificados distintas serem necessárias na rede de grupos do GKM. É importante mencionar que é possível aumentar a resistência do VKM ao aumentar a conectividade da estrutura virtual, ou até mesmo solicitar que cada origem ache dois caminhos disjuntos dentro da estrutura virtual (apesar de não ter sido avaliado neste trabalho o impacto de tal modificação). Por outro lado o VKM é virtualmente 100% resistente a ataques de criação de identidade falsas, enquanto o GKM mantém os mesmos resultados apresentados no ataque de personificação. Isso ocorre por que no VKM os nós que fazem parte da estrutura virtual são diferenciados em relação a demais nós da rede. No GKM não há distinção entre os nós que fazem parte dos grupos.

Além disso, o VKM pode se comportar de forma similar ao *PGP-Like*, como o VKM-PA, apenas alterando um simples parâmetro. Para provar isto, o VKM também foi avaliado em ambientes sem qualquer tipo de ataque e em ambientes com ataque de falta de cooperação. Os resultados mostram que sob ataques de falta de cooperação, o VKM-PA tem um desempenho muito similar ao *PGP-Like*, sendo que os valores CE (tempo de convergência) e UR (conectividade dos nós) são praticamente iguais, tendo apenas uma razoável diferença com 80% de nós comprometidos, devido à desconexão da estrutura virtual. Portanto, o VKM pode suprir a necessidade de qualquer usuário com sua capacidade de se comportar de maneiras distintas dinamicamente sem nenhuma reinicialização ou reconfiguração da rede.

Um motivo para se ter um sistema de gerenciamento de chaves em MANETs é adicionar segurança aos protocolos de roteamento e assim, garantir o bom desempenho dos mesmos, que são essenciais para o funcionamento desse tipo de rede. Para garantir segurança na camada de rede, deve-se combinar esquemas de criptografia com protocolos de roteamento. Como o VKM utiliza uma estrutura virtual para estabelecer a confiança entre os nós, foi apresentado um estudo no impacto ao roteamento de protocolos virtuais quando os mesmos incorporam o VKM. Os protocolos *Virtual Routing Protocol* (VRP) e

o *Virtual Distance Vector* (VDV) são protocolos híbridos que fazem uso de uma estrutura virtual para realizar a parte pró-ativa do protocolo. Assim, os mesmos não necessitam de qualquer pré-requisito para incorporar o VKM. Portanto, os mesmos foram escolhidos para comparação justamente por serem protocolos híbridos que fazem uso de virtualização, da mesma forma que o esquema de gerenciamento de chaves apresentado nesse trabalho.

O estudo do impacto ao roteamento desses protocolos levou em consideração a desempenho da taxa de entrega, a sobrecarga causada na rede pelo protocolo, e o atraso no envio de dados. A comparação foi realizada entre a especificação original de cada protocolo e a sua versão com o VKM incorporado.

No VRP e no VDV as rotas são obtidas pró-ativamente seguindo as regras definidas pela estrutura virtual. No VKM os nós seguem a confiança estabelecida por essa estrutura. Portanto é possível manter as mensagens de dados trafegando apenas pelas rotas adquiridas pró-ativamente, seguindo o modelo de confiança transitiva da estrutura virtual. Para evitar que a comunicação fim-a-fim entre dois nós não siga esse caminho virtual para entregar mensagens, foi removida a descoberta de rota através de *floodings*.

No VRP, a remoção das mensagens de RREQs na rede tem um pequeno impacto na taxa de entrega de dados. Devido a necessidade de esperar as atualizações pró-ativas das rotas, o atraso no envio de dados pode ser triplicado. No VDV, a remoção das mensagens de RREQs na rede tem um drástico impacto na taxa de entrega de dados e em um aumento na sobrecarga causada na rede. O atraso no envio de mensagens no VDV é ligeiramente afetado e a sobrecarga não é afetada.

Os resultados obtidos ao comparar o VRP com o VRP/VKM (VRP com o VKM embutido) mostram que a requisição de certificados pela origem em uma comunicação fim-a-fim chegam a aumentar em até 7 vezes a sobrecarga causada na rede. Devido a isso, a taxa de entrega cai em todos os cenários simulados para menos de 60 %. Outro problema gerado é o atraso no envio de mensagens, que também chega a ser 8 vezes maior. Portanto, implementar o VKM no VRP aumenta a sobrecarga na rede, diminuindo a taxa de entrega e aumentando o atraso no envio de dados.

Os resultados obtidos ao comparar o VDV com o VDV/VKM (VDV com o VKM embutido) mostram que manter as mensagens de dados trafegando dentro do caminho virtual causa uma drástica queda na taxa de entrega. Em alguns ambientes simulados, a taxa de entrega ficou menor do que 20%. Ainda, devido ao aumento no envio de mensagens de RRERs, a sobrecarga pode aumentar até 4 vezes, e o atraso no envio de dados chega a ser até 2 vezes maior. Incorporar o VKM no VDV aumenta a sobrecarga na rede, diminuindo a taxa de entrega e aumentando o atraso no envio de dados, porém os resultados observados mostram que boa parte no impacto do desempenho do VDV é devido a natureza do protocolo, que depende do envio de mensagens de RREQ quando não é possível enviar o dado de um nó para o próximo salto no caminho virtual.

Esse trabalho apresentou portanto, um esquema de gerenciamento de chaves públicas mais seguro que o *PGP-Like* sob ataques de personificação e tão resistente quanto o mesmo em cenários sob ataques de falta de cooperação. Ainda, o novo esquema apresentou maior resistência contra ataques de criação de identidades falsas quando comparado com o GKM. O VKM mostrou-se ser de fácil implementação em protocolos de roteamento virtuais, mas o impacto ao roteamento observado nesse trabalho aponta que faz-se necessário um estudo sobre melhorias e otimizações a esses protocolos quando utilizam o VKM.

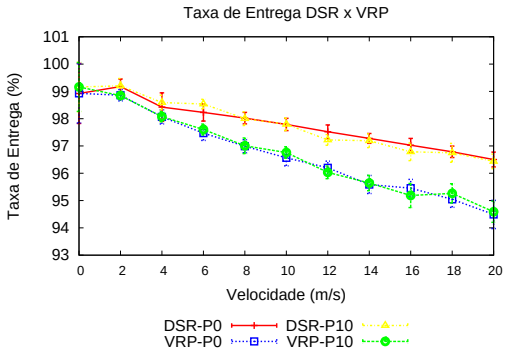
Trabalhos futuros incluem: comparar o desempenho do VKM com diferentes esquemas de gerenciamento de chaves existentes; avaliar o desempenho do VKM utilizando outras estruturas virtuais; avaliar o VKM em outros tipos de aplicações para MANETs; apresentar soluções, melhorias e otimizações para os protocolos de roteamento virtuais quando os mesmos incorporam o VKM como esquema de gerência de chaves e avaliar novamente o desempenho obtido utilizando essas modificações.

APÊNDICE A

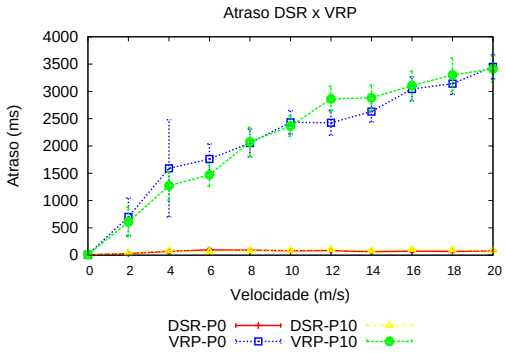
VALIDAÇÃO DO VRP

Este Apêndice apresenta os resultados obtidos na validação do protocolo VRP. A validação do VRP foi realizada comparando os resultados obtidos com a implementação do DSR no NS-2 que seguem as especificações descritas na RC4728 [28]. O VRP foi comparado com o DSR para validar os resultados obtidos neste trabalho de acordo com os resultados obtidos em [2]. Os parâmetros usados nas simulações são os mesmos apresentados na Tabela 6.1. Os resultados são médias de trinta e cinco simulações com intervalo de confiança de 95%. A estrutura virtual escolhida foi o RoR com três anéis. Cada nó mantém rota pró-ativamente para 5 outros nós.

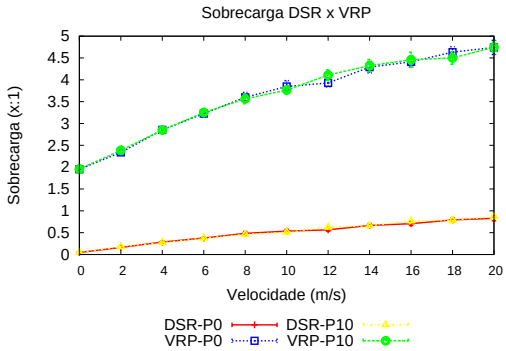
É possível observar que apesar dos resultados da taxa de entrega do VRP serem inferiores aos apresentados pelo DSR, a taxa de entrega do VRP se manteve sempre acima de 90 %, como apresentado em [2]. Igualmente, o atraso para o envio de mensagens e a sobrecarga gerada na rede foram similares aos apresentados em [2]. O VRP, quando comparado com o DSR em sua especificação original, apresenta uma taxa de entrega superior à taxa de entrega do DSR, o que não ocorreu nessa validação. Entretanto, como a implementação do DSR difere da usada no NS-2 (a qual apresenta diversas melhorias e adições para melhorar a taxa de entrega) em relação a versão usada em [2], o DSR apresenta resultados superiores aos obtidos na concepção original do VRP quando a comparação é realizada no simulador NS-2. De qualquer forma, a versão implementada do VRP segue as mesmas especificações apresentadas em [2] e os resultados absolutos do VRP (sem comparação com os resultados do DSR) são semelhantes aos publicados da mesma.



(a) Taxa de Entrega

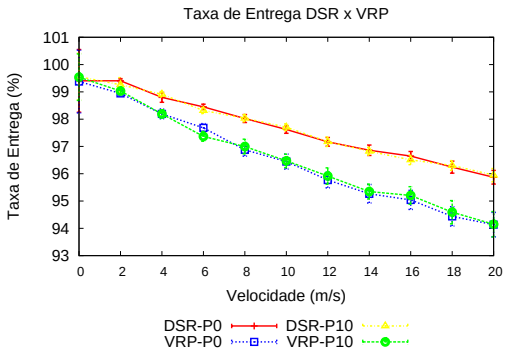


(b) Atraso

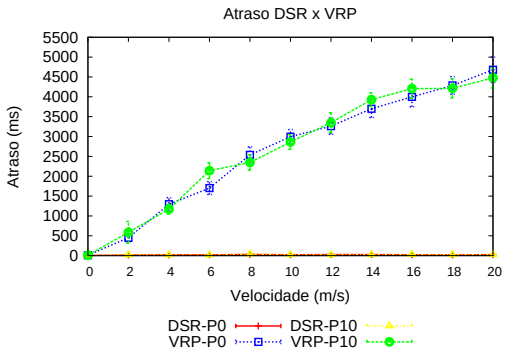


(c) Sobrecarga

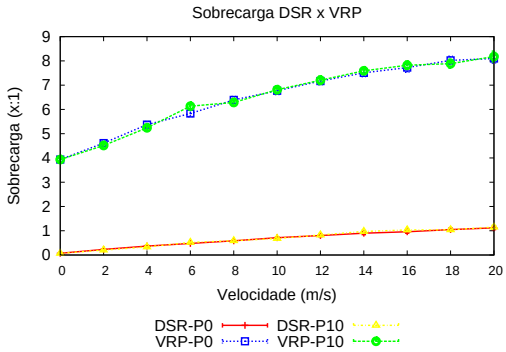
Figura A.1: DSR X VRP - Cenário 1000mx1000m com 51 nós



(a) Taxa de Entrega

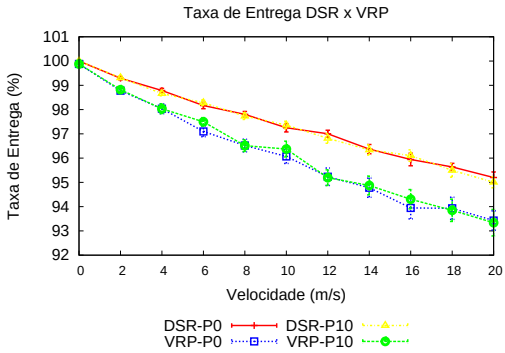


(b) Atraso

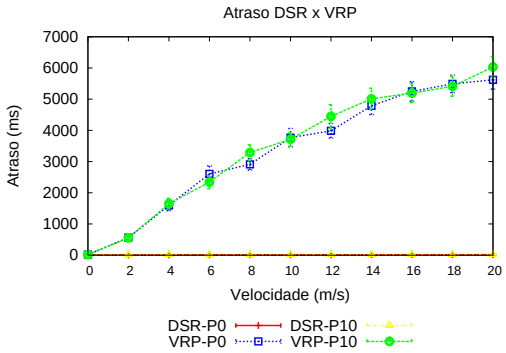


(c) Sobrecarga

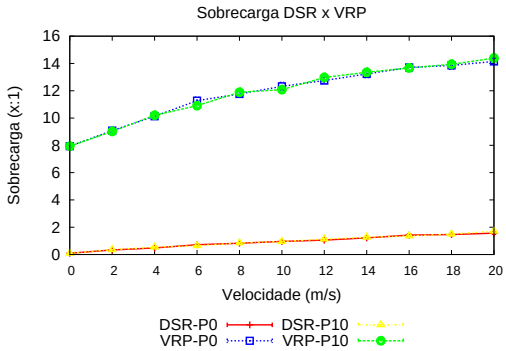
Figura A.2: DSR X VRP - Cenário 1000mx1000m com 75 nós



(a) Taxa de Entrega

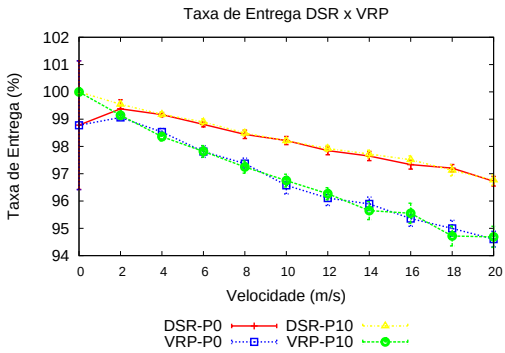


(b) Atraso

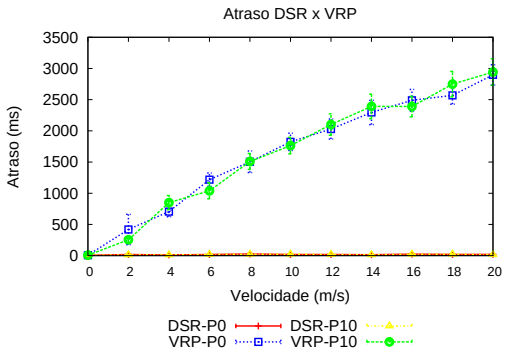


(c) Sobrecarga

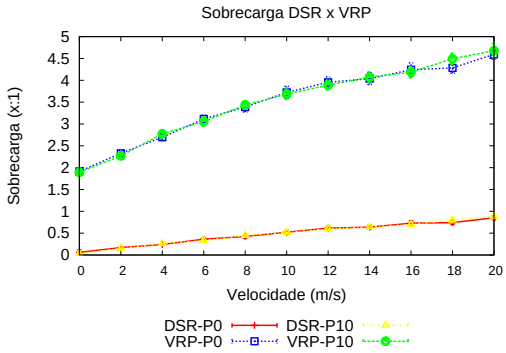
Figura A.3: DSR X VRP - Cenário 1000mx1000m com 108 nós



(a) Taxa de Entrega

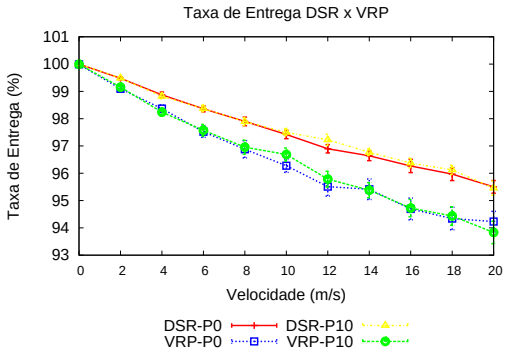


(b) Atraso

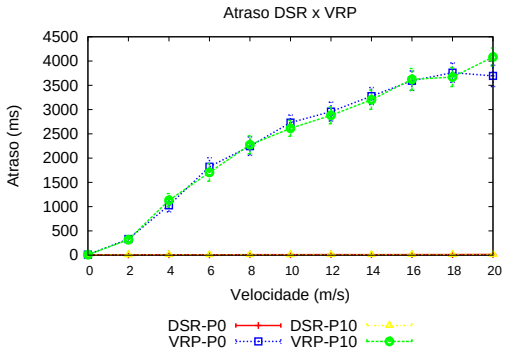


(c) Sobrecarga

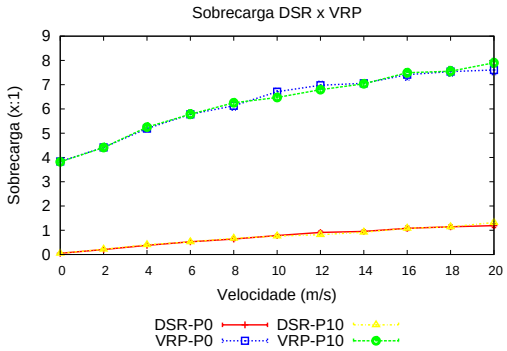
Figura A.4: DSR X VRP - Cenário 1500mx300m com 51 nós



(a) Taxa de Entrega

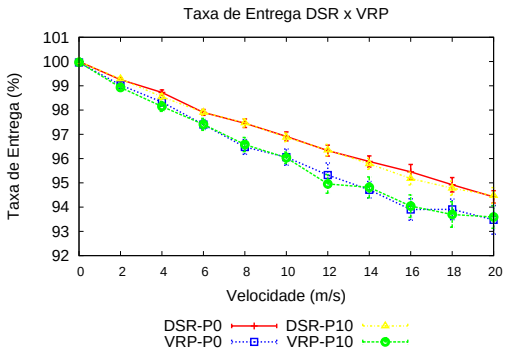


(b) Atraso

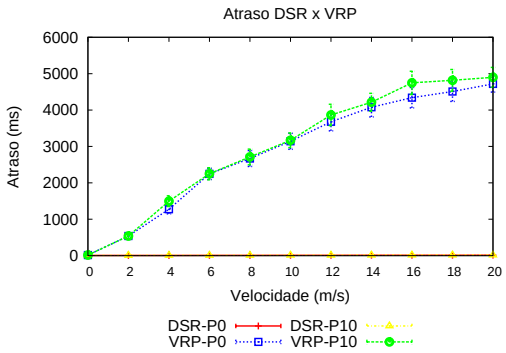


(c) Sobrecarga

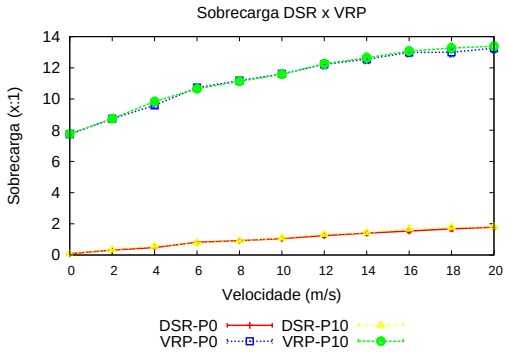
Figura A.5: DSR X VRP - Cenário 1500mx300m com 75 nós



(a) Taxa de Entrega



(b) Atraso



(c) Sobrecarga

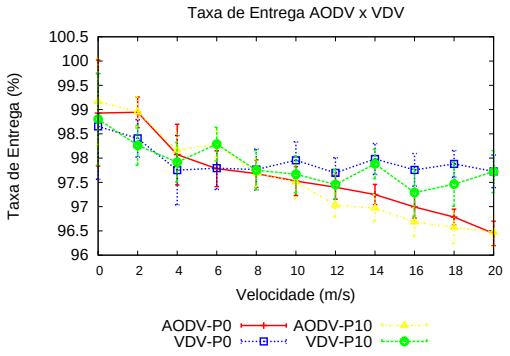
Figura A.6: DSR X VRP - Cenário 1500mx300m com 108 nós

APÊNDICE B

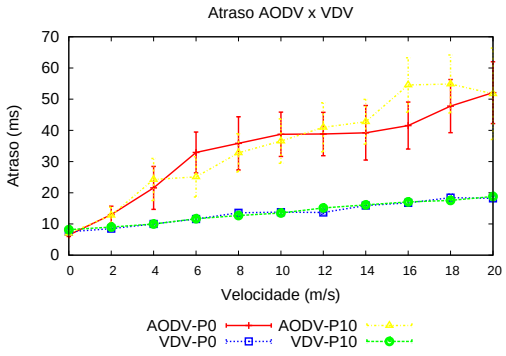
VALIDAÇAO DO VDV

Este Apêndice apresenta os resultados obtidos na validação do protocolo VDV. A validação do VDV foi realizada comparando os resultados obtidos com a implementação do AODV no NS-2 que seguem as especificações descritas na RFC3561 [38]. O VDV foi comparado com o AODV para validar os resultados obtidos neste trabalho de acordo com os resultados obtidos em [40]. Os parâmetros usados nas simulações são os mesmos apresentados na Tabela 6.1. Os resultados são médias de trinta e cinco simulações com intervalo de confiança de 95%. A estrutura virtual escolhida foi o RoR com três anéis. Cada nó mantém rota pró-ativamente para 5 outros nós.

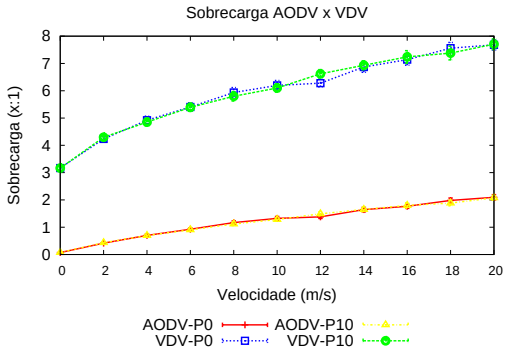
Como especificado na definição do VDV [40], enquanto a rede não se encontrar congestionada (com mais de 75 unidades na rede), o VDV é capaz de acelerar o envio de mensagens de dados. Entretanto, apesar de manter uma taxa de entrega superior ao AODV em quase todos os cenários simulados (com exceção do cenário com 108 nós espalhados em uma rede de 1000x1000m), o VDV apresenta uma sobrecarga até 5 vezes maior do que a do AODV e um atraso no envio de dados maior do que o apresentado pelo AODV quando a rede possui mais de 75 unidades na rede.



(a) Taxa de Entrega

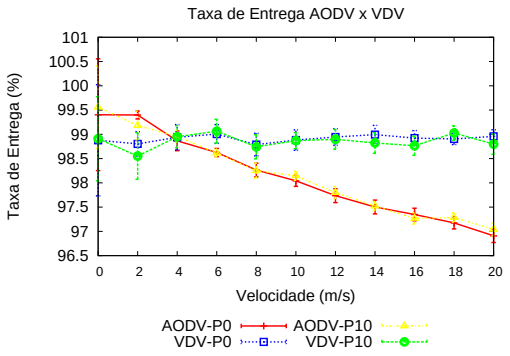


(b) Atraso

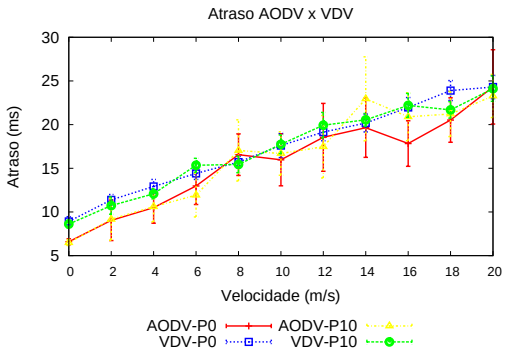


(c) Sobrecarga

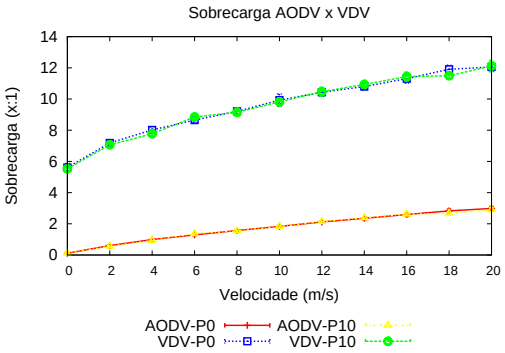
Figura B.1: AODV X VDV - Cenário 1000mx1000m com 51 nós



(a) Taxa de Entrega

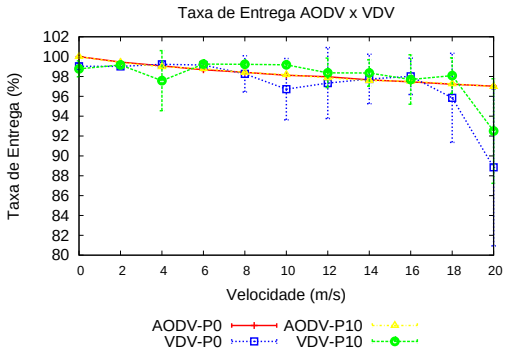


(b) Atraso

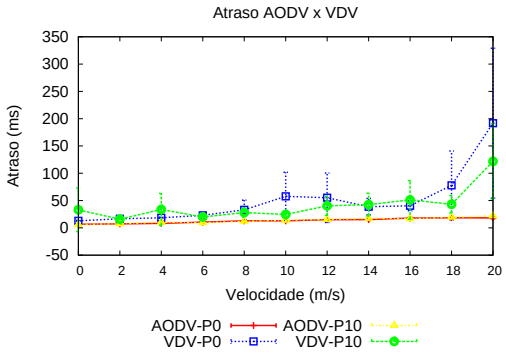


(c) Sobrecarga

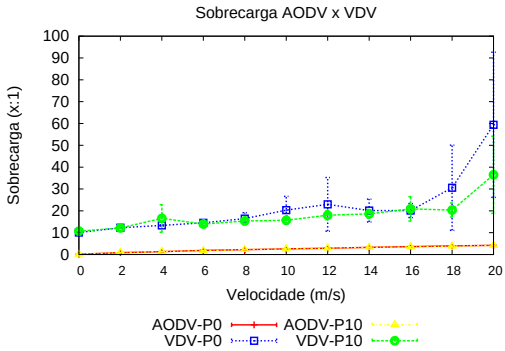
Figura B.2: AODV X VDV - Cenário 1000mx1000m com 75 nós



(a) Taxa de Entrega

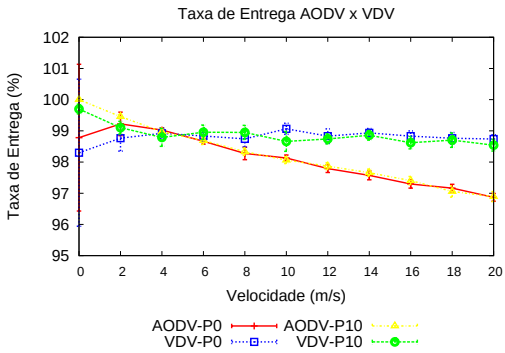


(b) Atraso

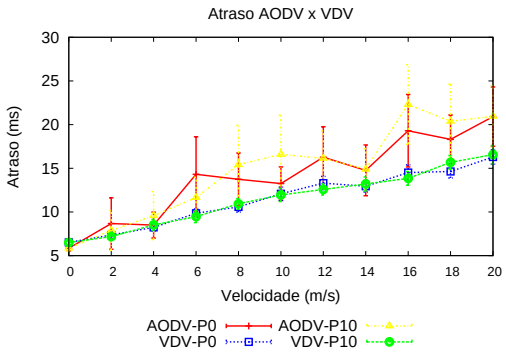


(c) Sobrecarga

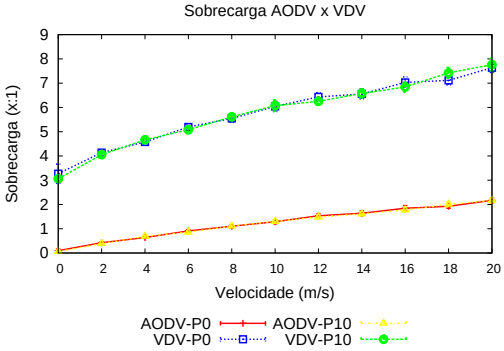
Figura B.3: AODV X VDV - Cenário 1000mx1000m com 108 nós



(a) Taxa de Entrega

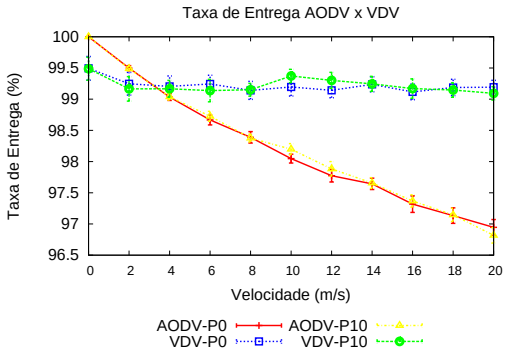


(b) Atraso

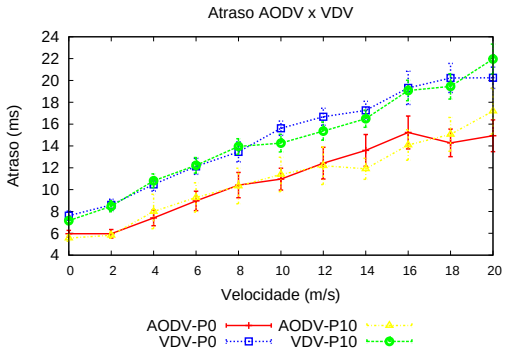


(c) Sobrecarga

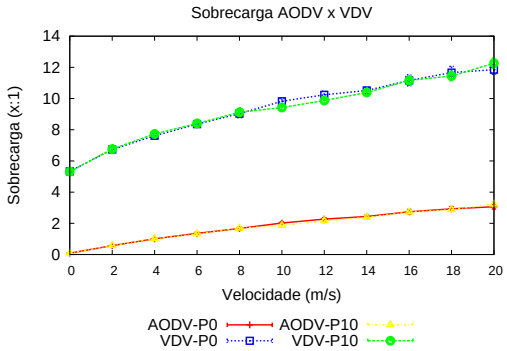
Figura B.4: AODV X VDV - Cenário 1500mx300m com 51 nós



(a) Taxa de Entrega

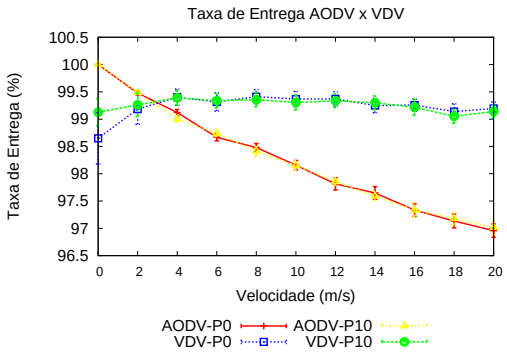


(b) Atraso

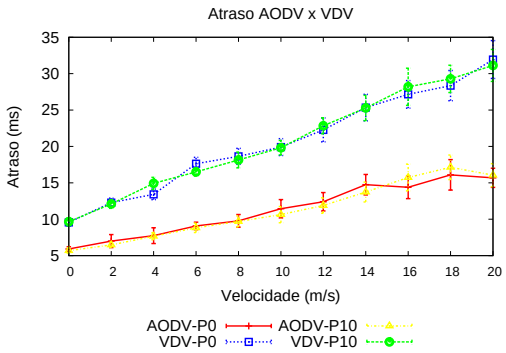


(c) Sobrecarga

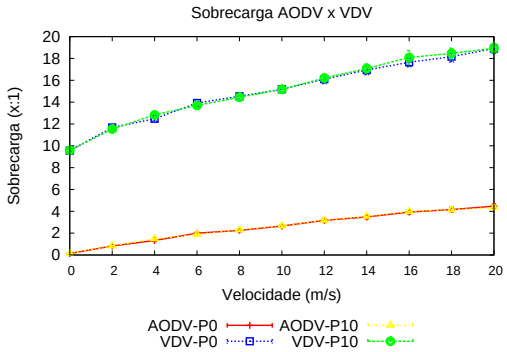
Figura B.5: AODV X VDV - Cenário 1500mx300m com 75 nós



(a) Taxa de Entrega



(b) Atraso



(c) Sobrecarga

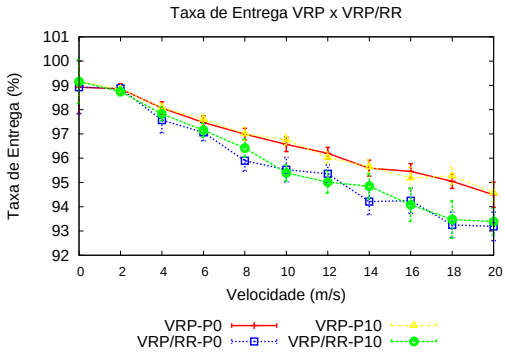
Figura B.6: AODV X VDV - Cenário 1500mx300m com 108 nós

APÊNDICE C

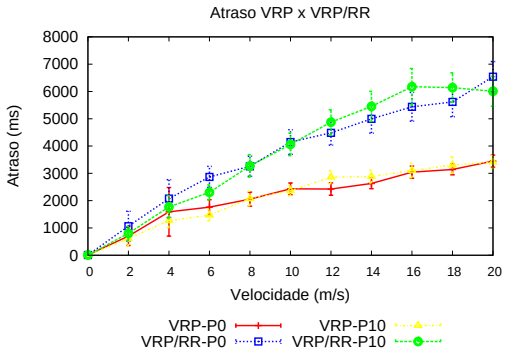
REMOÇÃO DA MENSAGEM DE RREQ DO PROTOCOLO VRP

Este Apêndice apresenta os resultados obtidos nas simulações do protocolo VRP quando o mesmo não utiliza a mensagem de RREQ na rede. Os parâmetros usados nas simulações são os mesmos apresentados na Tabela 6.1. Os resultados são médias de trinta e cinco simulações com intervalo de confiança de 95%. A estrutura virtual escolhida foi o RoR com três anéis. Cada nó mantém rota pró-ativamente para 5 outros nós.

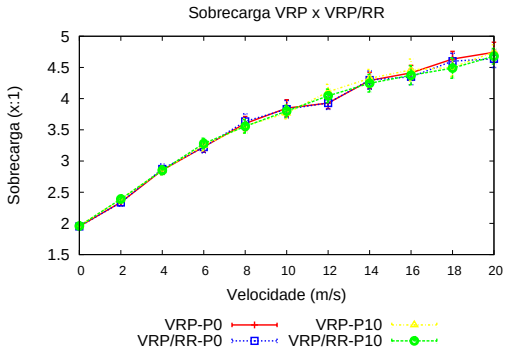
Remover as mensagens de RREQ no VRP não compromete a taxa de entrega do protocolo. Independente do número de unidades na rede, ou da velocidade das mesmas, a taxa de entrega se manteve acima de 90%. Isso ocorre por que o protocolo não depende exclusivamente da mensagem de RREQ para funcionar. Na maioria dos casos, as rotas são descobertas durante o funcionamento da parte pró-ativa do protocolo. Por outro lado o atraso para se obter uma rota para um destino pode chegar a ser 3 vezes maior, dependendo do número de unidades na rede. Isso ocorre pois somente em situações onde não foi possível descobrir uma rota para um destino utilizando a parte pró-ativa do protocolo, não há mais o *flooding* na rede com as mensagens de RREQ. Dessa forma um pacote deve aguardar até que uma nova rota seja descoberta de forma pró-ativa, ou que o mesmo seja descartado após expirar o seu tempo máximo para permanecer na fila. Mesmo com o aumento do atraso, não é observado um aumento da sobrecarga na comunicação dos nós.



(a) Taxa de Entrega

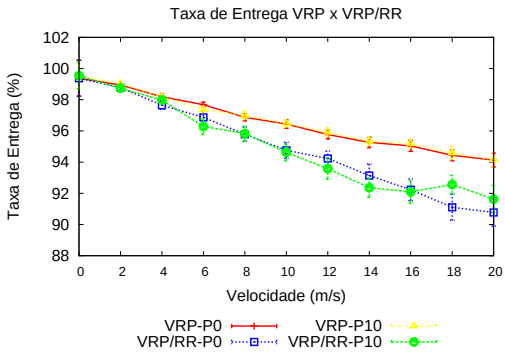


(b) Atraso

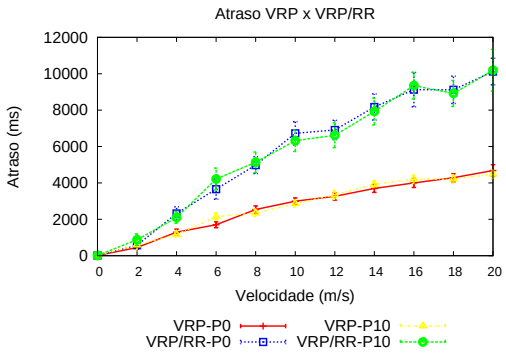


(c) Sobrecarga

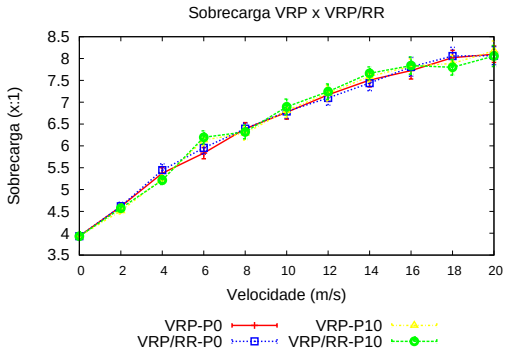
Figura C.1: VRP X VRP/RR - Cenário 1000mx1000m com 51 nós



(a) Taxa de Entrega

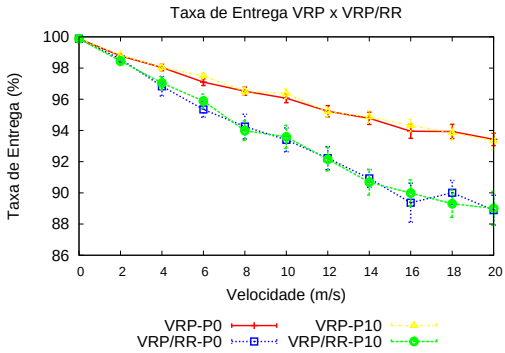


(b) Atraso

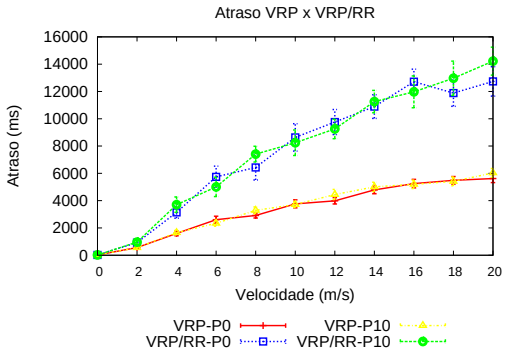


(c) Sobrecarga

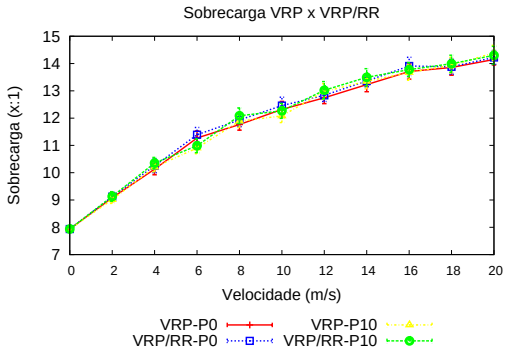
Figura C.2: VRP X VRP/RR - Cenário 1000mx1000m com 75 nós



(a) Taxa de Entrega

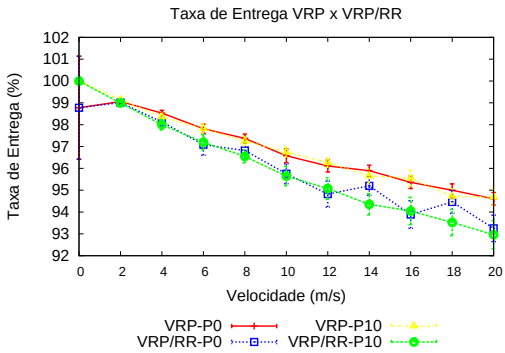


(b) Atraso

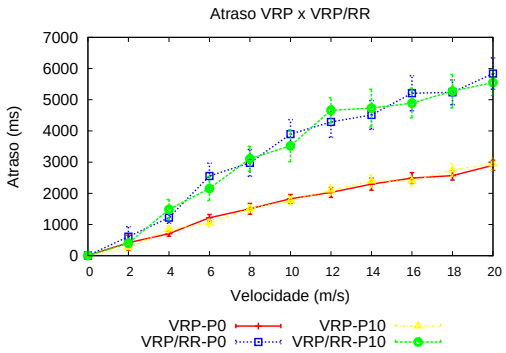


(c) Sobrecarga

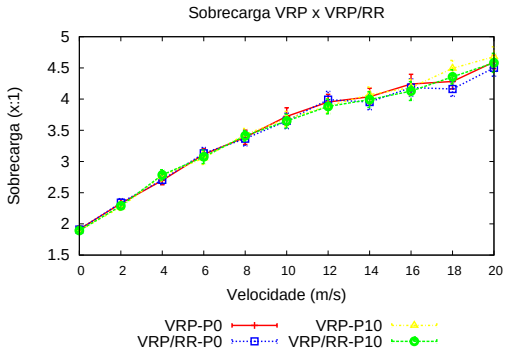
Figura C.3: VRP X VRP/RR - Cenário 1000mx1000m com 108 nós



(a) Taxa de Entrega

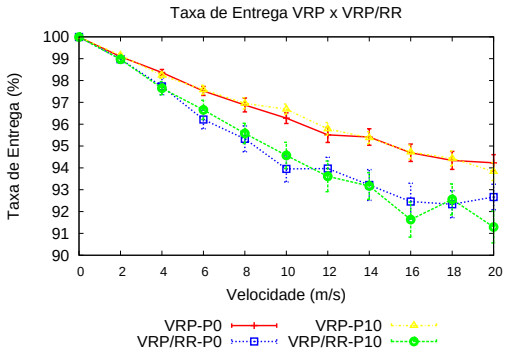


(b) Atraso

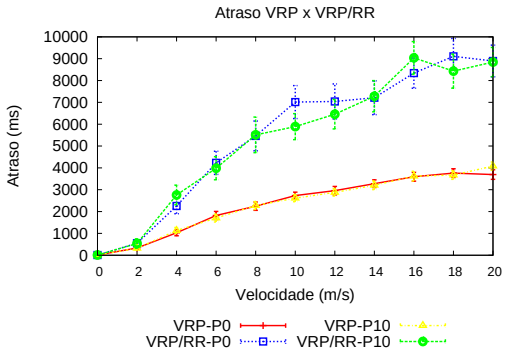


(c) Sobrecarga

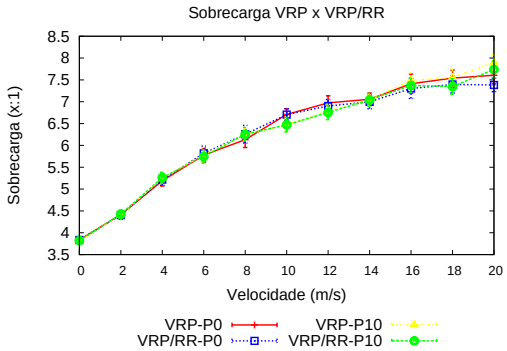
Figura C.4: VRP X VRP/RR - Cenário 1500mx300m com 51 nós



(a) Taxa de Entrega

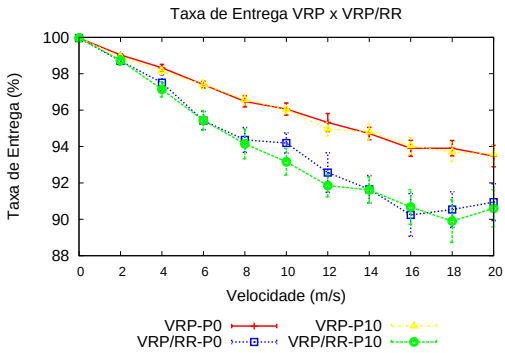


(b) Atraso

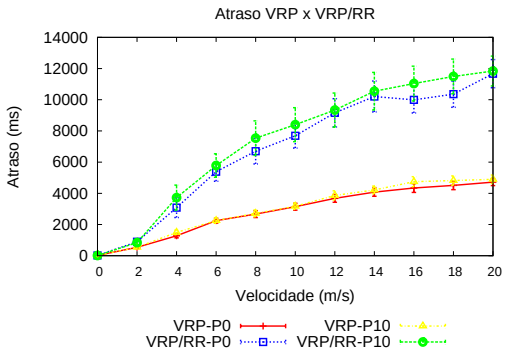


(c) Sobrecarga

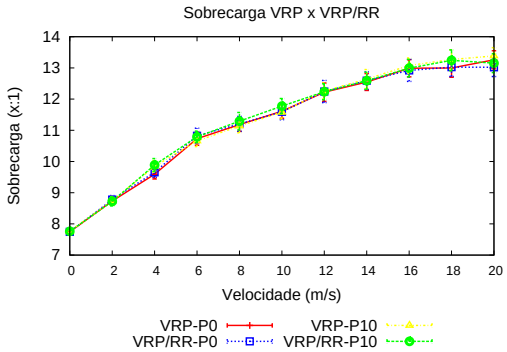
Figura C.5: VRP X VRP/RR - Cenário 1500mx300m com 75 nós



(a) Taxa de Entrega



(b) Atraso



(c) Sobrecarga

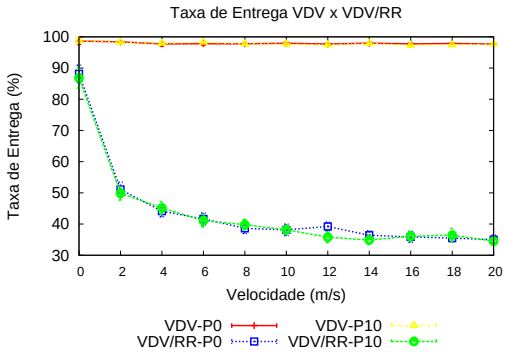
Figura C.6: VRP X VRP/RR - Cenário 1500mx300m com 108 nós

APÊNDICE D

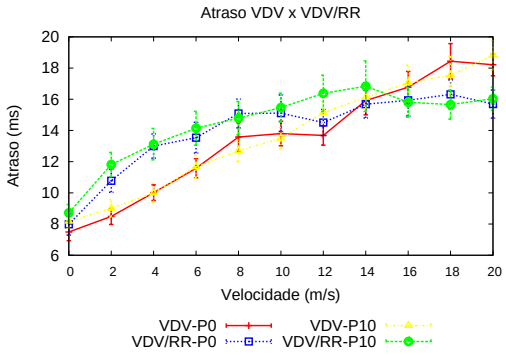
REMOÇÃO DA MENSAGEM DE RREQ DO PROTOCOLO VDV

Este Apêndice apresenta os resultados obtidos nas simulações do protocolo VDV quando o mesmo não utiliza a mensagem de RREQ na rede. Os parâmetros usados nas simulações são os mesmos apresentados na Tabela 6.1. Os resultados são médias de trinta e cinco simulações com intervalo de confiança de 95%. A estrutura virtual escolhida foi o RoR com três anéis. Cada nó mantém rota pró-ativamente para 5 outros nós.

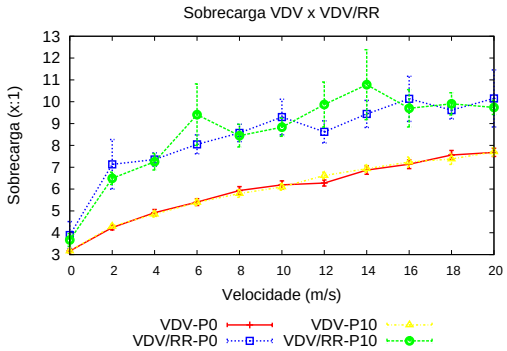
Remover as mensagens de RREQ no VDV compromete a taxa de entrega do protocolo. Independente do número de unidades na rede, ou da velocidade das mesmas, a taxa de entrega cai para menos de 60%. Isso ocorre por que o protocolo depende exclusivamente da mensagem de RREQ para funcionar. A funcionamento básico do protocolo consiste em mandar os pacotes de dados confiando na parte pró-ativa do protocolo. Porém, ao enviar dados sem haver uma mensagem de RREP por parte do destino, impede que a origem tenha certeza de que a rota realmente está disponível. Quando uma origem envia dados para um destino sem ter certeza da existência de uma rota para o mesmo, ela confia que os nós intermediários serão capazes de estabelecer uma nova rota caso a parte pró-ativa do protocolo falhe. Remover as mensagens de RREQ, portanto, altera completamente o funcionamento do VDV. Por outro lado o atraso para o envio de dados permanece similar pois pacotes que conseguem chegar ao destino ainda são enviados de maneira acelerada. Por fim é observado um aumento da sobrecarga na comunicação dos nós devido ao envio de mensagens de erro e ao aumento no envio de mensagens de atualização de rota.



(a) Taxa de Entrega

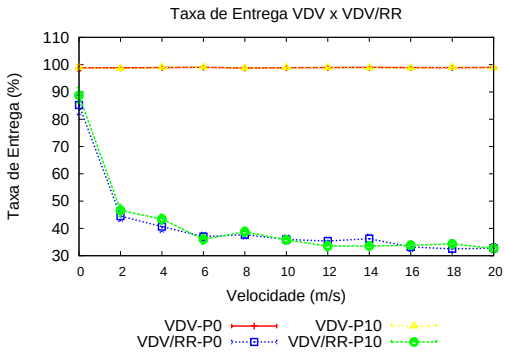


(b) Atraso

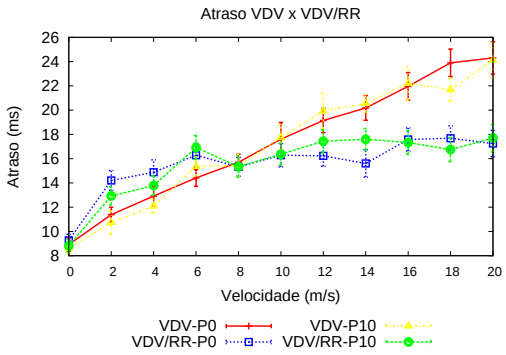


(c) Sobrecarga

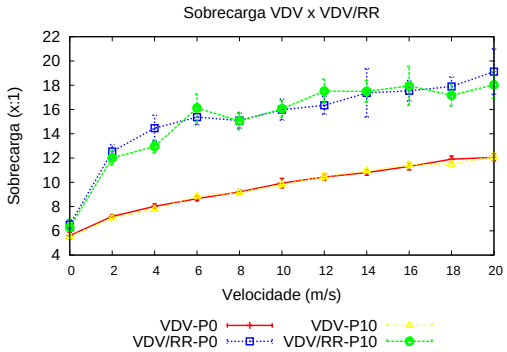
Figura D.1: VDV X VDV/RR - Cenário 1000mx1000m com 51 nós



(a) Taxa de Entrega

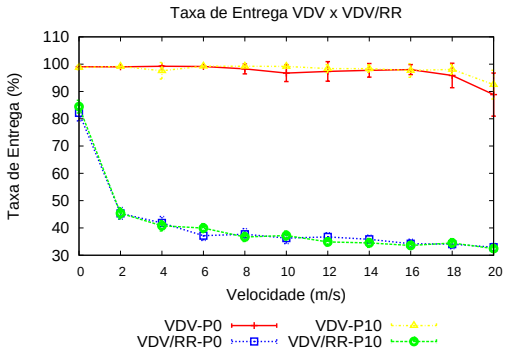


(b) Atraso

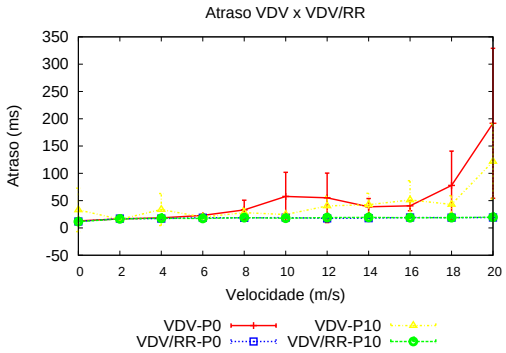


(c) Sobrecarga

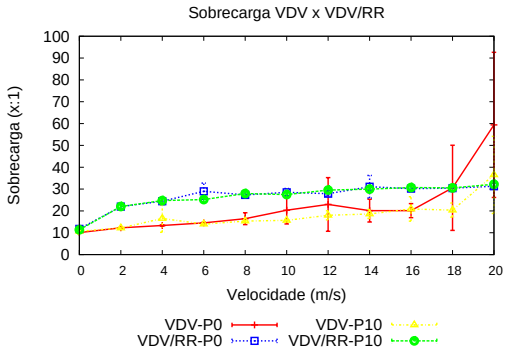
Figura D.2: VDV X VDV/RR - Cenário 1000mx1000m com 75 nós



(a) Taxa de Entrega

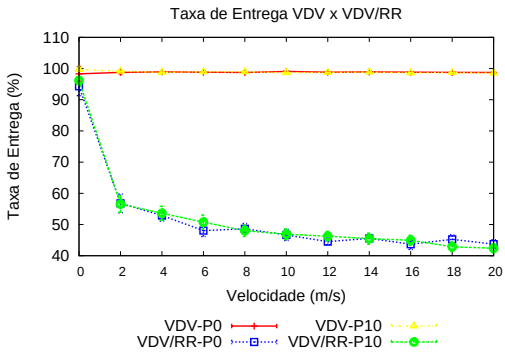


(b) Atraso

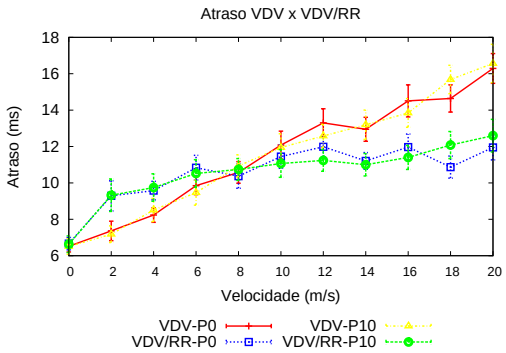


(c) Sobrecarga

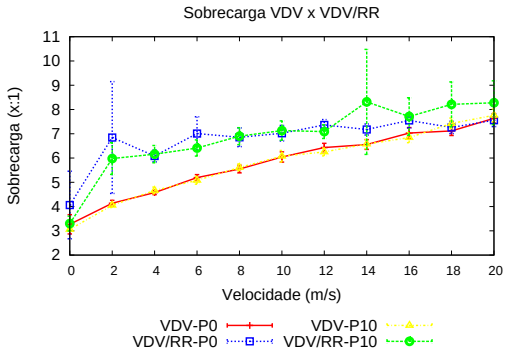
Figura D.3: VDV X VDV/RR - Cenário 1000mx1000m com 108 nós



(a) Taxa de Entrega

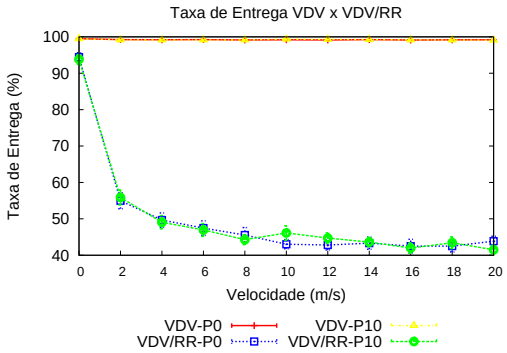


(b) Atraso

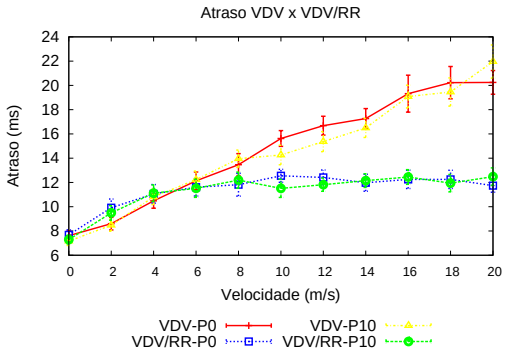


(c) Sobrecarga

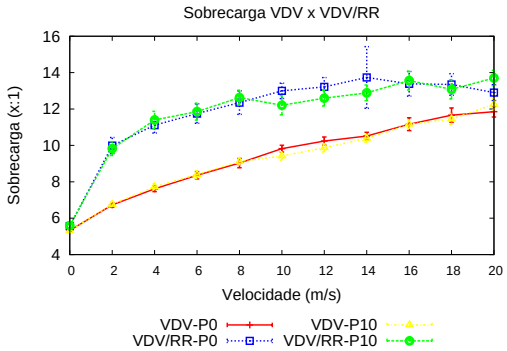
Figura D.4: VDV X VDV/RR - Cenário 1500mx300m com 51 nós



(a) Taxa de Entrega

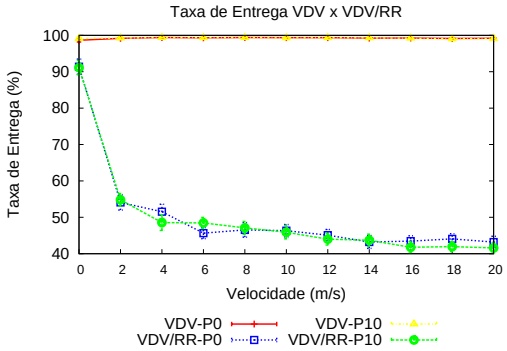


(b) Atraso

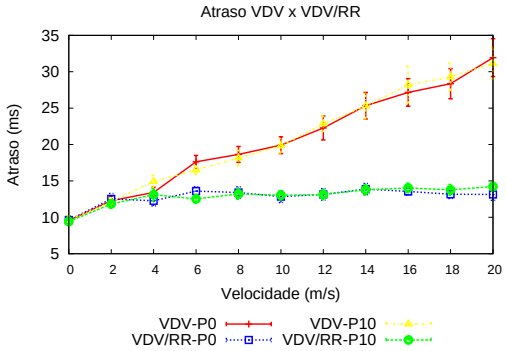


(c) Sobrecarga

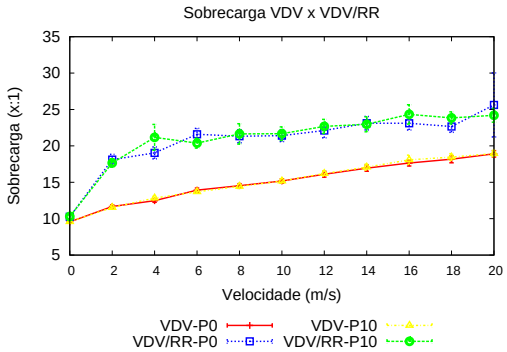
Figura D.5: VDV X VDV/RR - Cenário 1500mx300m com 75 nós



(a) Taxa de Entrega



(b) Atraso



(c) Sobrecarga

Figura D.6: VDV X VDV/RR - Cenário 1500mx300m com 108 nós

APÊNDICE E

LISTA DE PUBLICAÇÕES

Os estudos, implementações e testes realizados durante a elaboração deste trabalho renderam duas publicações:

1. SILVA, Renan Fischer et al[17]: artigo sobre o novo esquema de gerenciamento de chaves proposto. Esse artigo foi publicado e apresentado na **International Conference on Security and Cryptography (SECRYPT) 2009**, em Milão, Itália.

2. SILVA, Renan Fischer *et.al* [18]: artigo sobre o novo esquema de gerenciamento de chaves apresentado nessa dissertação, complementando o trabalho publicado e apresentado na SECRYPT 2009. Esse artigo foi publicado e apresentado no **X Workshop de Testes e Tolerância a Falhas (WTF 2009)**, em João Pessoa.

BIBLIOGRAFIA

[1] *Data and Computer Communications (8th Edition).* Prentice Hall, 8th edition, Agosto de 2006.

[2] L.C.P. Albini, A. Caruso, S. Chessa, e P. Maestrini. Reliable routing in wireless ad hoc networks: The virtual routing protocol. *Journal of Network and Systems Management*, 14(3):335–358, Setembro de 2006.

[3] Patroklos Argyroudis e Donal O'Mahony. Secure Routing for Mobile Ad hoc Networks. *IEEE Communications Surveys and Tutorials*, 7(3):2–21, 3ºTrimestre de 2005.

[4] Angelo Bannack e Luiz C. P. Albini. Aplicando gestão de energia ao protocolo de roteamento para redes ad hoc móveis vrp. *Dissertação de Mestrado, Universidade Federal do Paraná*, 2008.

[5] Angelo Bannack, Eduardo da Silva, Michele N. Lima, Aldri L. dos Santos, e Luiz C. P. Albini. Segurança em redes ad hoc. *In XXVI Simpósio Brasileiro de Telecomunicações (SBrT) 2008*, Setembro de 2008.

[6] Rajendra V. Boppana. e Satyadeva P. Konduru. An adaptive distance vector routing algorithm for mobile, ad hoc networks. *INFOCOM '2001: Proceedings of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, páginas 1753–1762, Abril de 2001.

[7] Srdjan Čapkun, Levente Buttyán, e Jean-Pierre Hubaux. Self-organized public-key management for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(1):52–64, 2003.

[8] Srdjan Čapkun, Jean-Pierre Hubaux, e Levente Buttyán. Mobility helps security in ad hoc networks. *MobiHoc '03: Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing*, páginas 46–56, New York, NY, EUA, 2003. ACM Press.

[9] Srdjan Čapkun, Jean-Pierre Hubaux, e Levente Buttyán. Mobility helps peer-to-peer security. *IEEE Transactions on Mobile Computing*, 5(1):43–51, 2006.

[10] Alice Cheng e Eric Friedman. Sybilproof reputation mechanisms. *Proceedings of the 3rd Workshop on Economics of Peer-to-Peer Systems (P2P-Econ '05)*, páginas 128–132, New York, NY, EUA, 2005. ACM.

[11] B. Christianson. Why isn't trust transitive. *Proceedings of the International Workshop on Security Protocols (WSP 1996)*. IEEE Computer Society, 1996.

[12] Eduardo da Silva, Aldri L. dos Santos, e Luiz C. P. Albini. Gerenciamento de chaves públicas sobrevivente baseado em grupos para manets. *Dissertação de Mestrado, Universidade Federal do Paraná*, 2009.

[13] George Danezis e Prateek Mittal. Sybilinfer: Detecting sybil nodes using social networks. *Proceedings of 16th Annual Network & Distributed System Security Symposium (NDSS '09)*, Fevereiro de 2009.

[14] Whitfield Diffie e Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, Novembro de 1976.

[15] Djamel Djenouri, Lyes Khelladi, e Ndjib Badache. A survey of security issues in mobile ad hoc and sensor networks. *IEEE Surveys and Tutorials*, 7(4):2–28, 2005.

[16] John R. Douceur. The sybil attack. *Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS 01)*, páginas 251–260, 2001.

[17] Fischer e Silva, Eduardo Silva, e Luiz Carlos Pessoa Albini. Resisting impersonation attacks in chaining-based public-key management on manets: the virtual public-key management. *Proceedings of the International Conference on Security and Cryptography (SECRYPT 2009)*, páginas 155–158, Julho de 2009.

[18] Renan Fischer e Silva, Eduardo Silva, e Luiz Carlos Pessoa Albini. Resistindo a ataques de personificação no gerenciamento de chaves públicas em redes ad hoc

móveis: Virtual public-key management system. *X Workshop de Testes e Tolerância a Falhas (WTF 2009)*, Agosto de 2009.

[19] Laurent Eschenauer e Virgil D. Gligor. A key-management scheme for distributed sensor networks. *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002)*, páginas 41–47, New York, NY, EUA, 2002. ACM Press.

[20] Nishu Garg e R.P.Mahapatra. Manet security issues. *IJCSNS: International Journal of Computer Science and Network Security*, volume 9, Agosto de 2009.

[21] Zygmunt J. Haas e Marc R. Pearlman. The zone routing protocol (zrp) for ad hoc networks. Internet-draft, IETF MANET Working Group, Novembro de 1997.

[22] Anne Marie Hegland, Eli Winjum, Stig F. Mjolsnes, Chunmig Rong, Oivind Kure, e Pal Spilling. A survey of key management in ad hoc networks. *IEEE Communications Surveys*, 08(03):48–66, 3ºTrimestre de 2006.

[23] Yih-Chun Hu, David B. Johnson, e Adrian Perrig. Sead: Secure efficient distance vector routing for mobile wireless ad hoc networks. *IEEE Workshop on Mobile Computing Systems and Applications*, 0:3, 2002.

[24] Yih-Chun Hu, Adrian Perrig, e David B. Johnson. Ariadne: a secure on-demand routing protocol for ad hoc networks. *MobiCom '02: Proceedings of the 8th Annual International Conference on Mobile Computing and Networking*, páginas 12–23, Nova York, NY, EUA, 2002. ACM.

[25] Jean-Pierre Hubaux, Levente Buttyán, e Srdan Čapkun. The quest for security in mobile ad hoc networks. *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc 2001)*, páginas 146–155, 2001.

[26] Kevin R. Hutson, Terri L. Schlosser, e Douglas R. Shier. On the distributed bellman-ford algorithm and the looping problem. *Informs Journal On Computing*, volume 19, páginas 542–551, Outono de 2007.

[27] David B. Johnson e David A. Maltz. Dynamic source routing in ad hoc wireless networks. *In Mobile Computing*, páginas 153–181, 1996.

[28] David B. Johnson e David A. Maltz. Rfc 4728 - the dynamic source routing protocol (dsr). *http://www.ietf.org/rfc/rfc4728.txt*, páginas 153–181, Fevereiro de 2007.

[29] Aram Khalili, Jonathan Katz, e William A. Arbaugh. Toward secure key distribution in truly ad-hoc networks. *Proceedings of the 2003 Symposium on Applications and the Internet Workshops (SAINT 2003 Workshops)*, páginas 342, Washington, DC, EUA, 2003. IEEE Computer Society.

[30] Michele Nogueira Lima, Aldri Luiz dos Santos, e Guy Pujolle. A survey of survivability in mobile ad hoc networks. *IEEE Communications Surveys & Tutorials*, 11:66–77, Fevereiro de 2009.

[31] J. Lundberg. *Routing security in ad hoc networks.* LUNDBERG, J. Routing security in ad Hoc networks. Relatório Técnico Tik110. 501, Helsinki University of Technology, 2000.

[32] Edith C. H. Ngai e Michael R. Lyu. Trust and clustering based authentication services in mobile ad hoc networks. *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW 2004)*, páginas 582–587, Washington, DC, EUA, 2004. IEEE Computer Society.

[33] Edith C. H. Ngai, Michael R. Lyu, e Roland T. Chin. An authentication service against dishonest users in mobile ad hoc networks. *Aerospace Conference 2004*, volume 02, páginas 1275–1285. IEEE, Março de 2004.

[34] Michele Nogueira, Guy Pujolle, Eduardo da Silva, Aldri dos Santos, e Luiz Carlos P. Albini. Survivable keying for wireless ad hoc networks. *Proceedings of the 2009 IFIP/IEEE International Symposium on Integrated Network Management (IM '09)*, páginas 606–613, Washington, DC, EUA, Jun de 2009. IEEE Communications Society.

[35] NS-2. The network simulator - ns-2, 2007.

[36] Charles E. Perkins e Pravin Bhagwat. Highly dynamic destinatio-sequenced distance-vector routing (dsdv) for mobile computers. *In SIGCOMM '94: Proceedings of the conference on Communications Architectures, Protocols and Applications*, volume 24, páginas 234–244, Outubro de 1994.

[37] Charles E. Perkins e Elizabeth M. Royer. Ad-hoc on-demand distance vector routing. *2nd IEEE Workshop on In Mobile Computing Systems and Applications*, páginas 90–100, 1999.

[38] Charles E. Perkins e Elizabeth M. Royer. Ad-hoc on-demand distance-vector (aodv) protocol. *http://www.ietf.org/rfc/rfc3561.txt*, páginas 90–100, Julho de 2003.

[39] Chris Piro, Clay Shields, e Brian Neil Levine. Sybilinfer: Detecting sybil nodes using social networks. *Proceeding of the IEEE/ACM International Conference on Security and Privacy in Communication Networks (SecureComm '06)*, páginas 1–11, Agosto de 2006.

[40] A. Robba e P. Maestrini. Routing in mobile ad-hoc networks: The virtual distance vector protocol. *Proceedings of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2007)*, páginas 1–9, 2007.

[41] Eduardo Silva, Aldri Luiz dos Santos, Luiz Carlos Pessoa Albini, e Michele Nogueira Lima. Quantify misbehavior attacks against the self-organized public key management on manets. *Proceedings of the International Conference on Security and Cryptography (SECRYPT 2008)*, páginas 128–135, Julho de 2008.

[42] Nguyen Tran, Bonan Min, Jinyang Li, e Lakshminarayanan Subramanian. Sybil resilient online content voting. *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation (NSDI '09)*, páginas 15–28, Boston, MA, EUA, Abril de 2009. USENIX Association.

[43] Johann van der Merwe, Dawoud Dawoud, e Stephen McDonald. A survey on peer-to-peer key management for mobile ad hoc networks. *ACM Computing Survey*, 39(1):1, 2007.

[44] I. M. Vinogradov. *An Introduction to the Theory of Numbers*. Pergamon Press, Londres &Nova York, 1955.

[45] Shiuh-Jeng Wang, Yuh-Ren Tsai, e Chung-Wei Chen. Strategues averting sybil-type attacks based on the blom-scheme in ad hoc sensor networks. *Journal of Communications (JCM)*, volume 3, páginas 20–26, 2008.

[46] Bing Wu, Jianmin Chen, Jie Wu, e Mihaela Cardei. A survey on attacks and countermeasures in mobile ad hoc networks. Y. Xiao, X. Shen, e D.-Z. Du, editors, *Wireless network security*. Springer, 2006.

[47] Manel Guerrero Zapata e N. Asokan. Securing ad hoc routing protocols. *WiSE '02: Proceedings of the 1st ACM Workshop on Wireless Security (2002)*, páginas 1–10, 2002.

[48] Philip R. Zimmermann. *The official PGP user's guide*. MIT Press, Cambridge, MA, EUA, 1995.

RENAN FISCHER E SILVA

SISTEMA DE GERENCIAMENTO DE CHAVES PÚBLICAS BASEADO EM VIRTUALIZAÇÃO PARA REDES AD HOC MÓVEIS

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.
Orientador: Prof. Dr. Luiz Carlos Pessoa Albini

CURITIBA

2010